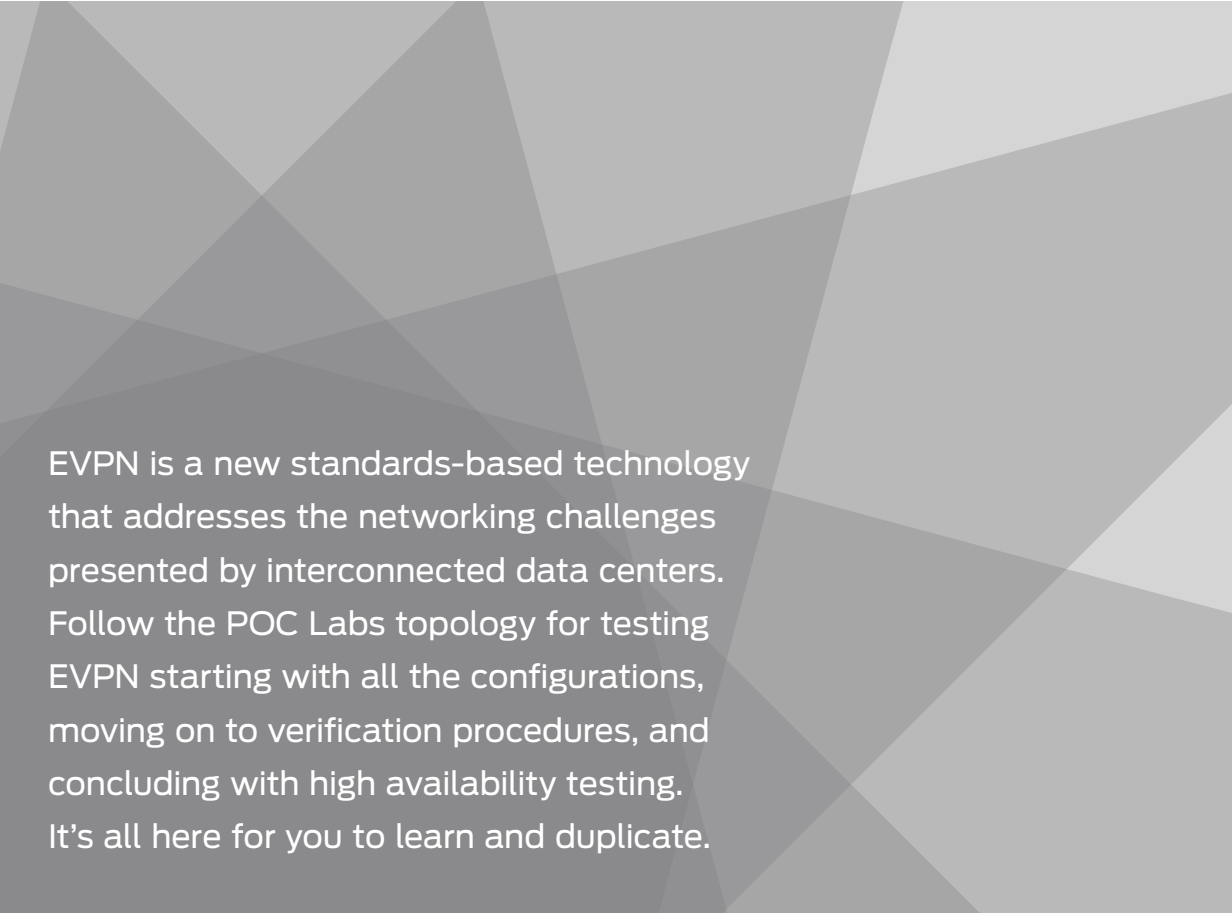


# DAY ONE: USING ETHERNET VPNS FOR DATA CENTER INTERCONNECT



EVPN is a new standards-based technology that addresses the networking challenges presented by interconnected data centers. Follow the POC Labs topology for testing EVPN starting with all the configurations, moving on to verification procedures, and concluding with high availability testing. It's all here for you to learn and duplicate.

By Victor Ganjian

# DAY ONE: USING ETHERNET VPNS FOR DATA CENTER INTERCONNECT

Today's virtualized data centers are typically deployed at geographically diverse sites in order to optimize the performance of application delivery to end users, and to maintain high availability of applications in the event of site disruption. Realizing these benefits requires the extension of Layer 2 connectivity across data centers, also known as Data Center Interconnect (DCI), so that virtual machines (VMs) can be dynamically migrated between the different sites. To support DCI, the underlying network is also relied upon to ensure that traffic flows to and from the VMs are forwarded along the most direct path, before, as well as after migration; that bandwidth on all available links is efficiently utilized; and, that the network recovers quickly to minimize downtime in the event of a link or node failure.

EVPN is a new technology that has attributes specifically designed to address the networking requirements of interconnected data centers. And *Day One: Using Ethernet VPNs for Data Center Interconnect* is a proof of concept straight from Juniper's Proof of Concept Labs (POC Labs). It supplies a sample topology, all the configurations, and the validation testing, as well as some high availability tests.

*"EVPN was recently published as a standard by IETF as RFC 7432, and a few days later it has its own Day One book! Victor Ganjian has written a useful book for anyone planning, deploying, or scaling out their data center business."*

**John E. Drake**, Distinguished Engineer, Juniper Networks, Co-Author of RFC 7432: EVPN

*"Ethernet VPN (EVPN) delivers a wide range of benefits that directly impact the bottom line of service providers and enterprises alike. However, adopting a new protocol is always a challenging task. This Day One book eases the adoption of EVPN technology by showing how EVPN's advanced concepts work and then supplying validated configurations that can be downloaded to create a working network. This is a must read for all engineers looking to learn and deploy EVPN technologies."*

**Sachin Natu**, Director, Product Management, Juniper Networks

Juniper Networks Books are singularly focused on network productivity and efficiency. Peruse the complete library at [www.juniper.net/books](http://www.juniper.net/books).

Published by Juniper Networks Books



**JUNIPER**  
NETWORKS

# Day One: Using Ethernet VPNs for Data Center Interconnect

By Victor Ganjian

<i>Chapter 1: About Ethernet VPNs.....</i>	<i>9</i>
<i>Chapter 2: Configuring EVPN.....</i>	<i>17</i>
<i>Chapter 3: Verification.....</i>	<i>37</i>
<i>Chapter 4: High Availability Tests.....</i>	<i>79</i>
<i>Conclusion.....</i>	<i>86</i>

© 2015 by Juniper Networks, Inc. All rights reserved. Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners. Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

**Published by Juniper Networks Books**

Author: Victor Ganjian

Technical Reviewers: Scott Astor, Ryan Bickhart, John E. Drake, Prasantha Gudipati, Russell Kelly, Matt Mellin, Brad Mitchell, Sachin Natu, Nitin Singh, Ramesh Yakkala

Editor in Chief: Patrick Ames

Copyeditor and Proofer: Nancy Koerbel

Illustrator: Karen Joice

J-Net Community Manager: Julie Wider

ISBN: 978-1-936779-04-6 (print)

Printed in the USA by Vervante Corporation.

ISBN: 978-1-936779-05-3 (ebook)

Version History: v1, March 2015

2 3 4 5 6 7 8 9 10

**About the Author:**

Victor Ganjian is currently a Senior Data Networking Engineer in the Juniper Proof of Concept lab in Westford, Massachusetts. He has 20 years of hands-on experience helping Enterprise and Service Provider customers understand, design, configure, test, and troubleshoot a wide range of IP routing and Ethernet switching related technologies. Victor holds B.S. and M.S. degrees in Electrical Engineering from Tufts University in Medford, Massachusetts.

**Author's Acknowledgments:**

I would like to thank all of the technical reviewers for taking the time to provide valuable feedback that significantly improved the quality of the content in this book.

I would like to thank Prasantha Gudipati in the Juniper System Test group and Nitin Singh and Manoj Sharma, the Technical Leads for EVPN in Juniper Development Engineering, for answering my EVPN-related questions via many impromptu conference calls and email exchanges as I was getting up to speed on the technology.

I would like to thank Editor in Chief Patrick Ames, copyeditor Nancy Koerbel, and illustrator Karen Joice for their guidance and assistance with the development of this book.

I would like to thank my colleagues in the Westford POC lab for their support and providing me with the opportunity to write this book.

Finally, I thank my family for their ongoing support, encouragement, and patience, allowing me the time and space needed to successfully complete this book.

This book is available in a variety of formats at:  
<http://www.juniper.net/dayone>.

## Welcome to Day One

This book is part of a growing library of *Day One* books, produced and published by Juniper Networks Books.

*Day One* books were conceived to help you get just the information that you need on day one. The series covers Junos OS and Juniper Networks networking essentials with straightforward explanations, step-by-step instructions, and practical examples that are easy to follow.

The *Day One* library also includes a slightly larger and longer suite of *This Week* books, whose concepts and test bed examples are more similar to a weeklong seminar.

You can obtain either series, in multiple formats:

- Download a free PDF edition at <http://www.juniper.net/dayone>.
- Get the ebook edition for iPhones and iPads from the iTunes Store. Search for Juniper Networks Books.
- Get the ebook edition for any device that runs the Kindle app (Android, Kindle, iPad, PC, or Mac) by opening your device's Kindle app and going to the Kindle Store. Search for Juniper Networks Books.
- Purchase the paper edition at either Vervante Corporation ([www.vervante.com](http://www.vervante.com)) for between \$12-\$28, depending on page length.
- Note that Nook, iPad, and various Android apps can also view PDF files.

## Audience

This book is intended for network engineers that have experience with other VPN technologies and are interested in learning how EVPN works to evaluate its use in projects involving interconnection of multiple data centers. Network architects responsible for designing EVPN networks and administrators responsible for maintaining EVPN networks will benefit the most from this text.

## What You Need to Know Before Reading This Book

Before reading this book, you should be familiar with the basic administrative functions of the Junos operating system, including the ability to work with operational commands and to read, understand, and change Junos configurations.

This book makes a few assumptions about you, the reader. If you don't meet these requirements the tutorials and discussions in this book may not work in your lab:

- You have advanced knowledge of how Ethernet switching and IP routing protocols work.
- You have knowledge of IP core networking and understand how routing protocols such as OSPF, MP-BGP, and MPLS are used in unison to implement different types of VPN services.
- You have knowledge of other VPN technologies, such as RFC 4364-based IP VPN and VPLS. IP VPN is especially important since many EVPN concepts originated from IP VPNs, and IP VPN is used in conjunction with EVPN in order to route traffic.

There are several books in the *Day One* library on learning Junos, and on MPLS, EVPN, and IP routing, at [www.juniper.net/dayone](http://www.juniper.net/dayone).

## What You Will Learn by Reading This Book

This *Day One* book will explain, in detail, the inner workings of EVPN. Upon completing it you will have acquired a conceptual understanding of the underlying technology and benefits of EVPN. Additionally, you will gain the practical knowledge necessary to assist with designing, deploying, and maintaining EVPN in your network with confidence.

## Get the Complete Configurations

The configuration files for all devices used in this POC Lab *Day One* book can be found on this book's landing page at <http://www.juniper.net/dayone>. The author has also set up a Dropbox download for those readers not logging onto the *Day One* website, at: <https://dl.dropboxusercontent.com/u/18071548/evpn-configs.zip>. Note that this URL is not under control of the author and may change over the print life of this book.

## Juniper Networks Proof of Concept (POC) Labs

Juniper Worldwide POC Labs are located in Westford, Mass. and Sunnyvale, California. They are staffed with a team of experienced network engineers that work with Field Sales Engineers and their customers to demonstrate specific features and test the performance of Juniper products. The network topologies and tests are customized for each customer based upon their unique requirements.

## Terminology

For your reference, or if you are coming from another vendor's equipment to Juniper Networks, a list of acronyms and terms pertaining to EVPN is presented below.

- BFD: Bidirectional Forwarding Detection, a simple Hello protocol that is used for rapidly detecting faults between neighbors or adjacencies of well-known routing protocols.
- BUM: Broadcast, unknown unicast, and multicast traffic. Essentially multi-destination traffic.
- DF: Designated Forwarder. The EVPN PE responsible for forwarding BUM traffic from the core to the CE.
- ES: Ethernet Segment. The Ethernet link(s) between a CE device and one or more PE devices. In a multi-homed topology the set of links between the CE and PEs is considered a single "Ethernet Segment." Each ES is assigned an identifier.
- ESI: Ethernet Segment Identifier. A 10 octet value with range from 0x00 to 0xFFFFFFFFFFFFFFFF which represents the ES. An ESI must be set to a network-wide unique, non-reserved

value when a CE device is multi-homed to two or more PEs. For a single homed CE the reserved ESI value 0 is used. The ESI value of “all FFs” is also reserved.

- EVI: EVPN Instance, defined on PEs to create the EVPN service.
- Ethernet Tag Identifier: Identifies the broadcast domain in an EVPN instance. For our purposes the broadcast domain is a VLAN and the Ethernet Tag Identifier is the VLAN ID.
- IP VPN - a Layer 3 VPN service implemented using BGP/MPLS IP VPNs (RFC 4364)
- LACP: Link Aggregation Control Protocol, used to manage and control the bundling of multiple links or ports to form a single logical interface.
- LAG: Link aggregation group.
- MAC-VRF: MAC address virtual routing and forwarding table. This is the Layer 2 forwarding table on a PE for an EVI.
- MP2MP: Multipoint to Multipoint.
- P2MP: Point to Multipoint.
- PMSI: Provider multicast service interface. A logical interface in a PE that is used to deliver multicast packets from a CE to remote PEs in the same VPN, destined to CEs.



# Chapter 1

## About Ethernet VPNs (EVPN)

*Ethernet VPN*, or simply EVPN, is a new standards-based technology that provides virtual multi-point bridged connectivity between different Layer 2 domains over an IP or IP/MPLS backbone network. Similar to other VPN technologies such as IP VPN and VPLS, EVPN instances (EVIs) are configured on PE routers to maintain logical service separation between customers. The PEs connect to CE devices, which can be a router, switch, or host over an Ethernet link. The PE routers then exchange reachability information using Multi-Protocol BGP (MP-BGP) and encapsulated customer traffic is forwarded between PEs. Because elements of the architecture are common with other VPN technologies, EVPN can be seamlessly introduced and integrated into existing service environments.

A unique characteristic of EVPN is that MAC address learning between PEs occurs in the control plane. A new MAC address detected from a CE is advertised by the local PE to all remote PEs using an MP-BGP MAC route. This method differs from existing Layer 2 VPN solutions such as VPLS, which performs MAC address learning by flooding unknown unicast in the data plane. This control plane-based MAC learning method provides a much finer control over the virtual Layer 2 network and is the key enabler of the many compelling features provided by EVPN that we will explore in this book.

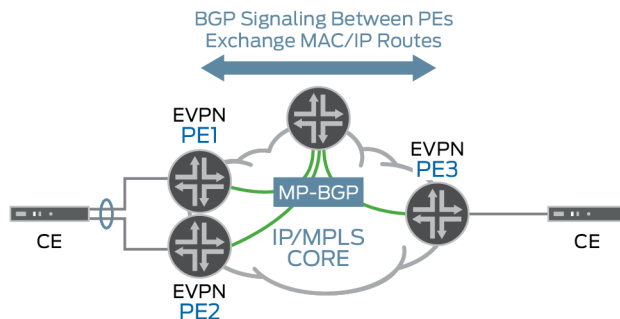


Figure 1.1 High Level View of EVPN Control Plane

Service Providers and Enterprises can use EVPN to implement and offer next-generation Layer 2 VPN services to their customers. EVPN has the flexibility to be deployed using different topologies including E-LINE, E-LAN, and E-TREE. It supports an all-active mode of multi-homing between the CE and PE devices that overcomes the limitations of existing solutions in the areas of resiliency, load balancing, and efficient bandwidth utilization. The control plane-based MAC learning allows a network operator to apply policies to control Layer 2 MAC address learning between EVPN sites and also provides many options for the type of encapsulation that can be used in the data plane.

EVPN's integrated routing and bridging (IRB) functionality supports both Layer 2 and Layer 3 connectivity between customer edge nodes along with built-in Layer 3 gateway functionality. By adding the MAC and IP address information of both hosts and gateways in MAC routes, EVPN provides optimum intra-subnet and inter-subnet forwarding within and across data centers. This functionality is especially useful for Service Providers that offer Layer 2 VPN, Layer 3 VPN, or Direct Internet Access (DIA) services and want to provide additional cloud computation and/or storage services to existing customers.

**MORE?** During the time this *Day One* book was being produced, the proposed BGP MPLS-Based Ethernet VPN draft specification was adopted as a standard by the IETF and published as RFC 7432. The document can be viewed at <http://tools.ietf.org/html/rfc7432/>. For more details on requirements for EVPN, visit: <http://tools.ietf.org/html/rfc7209>.

## EVPN for DCI

There is a lot of interest in EVPN today because it addresses many of the challenges faced by network operators that are building data centers to offer cloud and virtualization services. The main application of EVPN is Data Center Interconnect (DCI), the ability to extend Layer 2 connectivity between different data centers. Geographically diverse data centers are typically deployed to optimize the performance of application delivery to end users and to maintain high availability of applications in the event of site disruption.

Some of the DCI requirements addressed by EVPN include:

- Multi-homing between CE and PE with support for active-active links.
- Fast service restoration.
- Support for virtual machine (VM) migration or MAC Mobility.
- Integration of Layer 3 routing with optimal forwarding paths.
- Minimizing bandwidth utilization of multi-destination traffic between data center sites.
- Support for different data plane encapsulations.

### All-Active Multi-homing

EVPN supports *all-active* multi-homing, which allows a CE device to connect to two or more PE routers such that traffic is forwarded using all of the links between the devices. This enables the CE to load balance traffic to the multiple PE routers. More importantly it enables *Aliasing* which allows a remote PE to load balance traffic to the multi-homed PEs across the core network, even when the remote PE learns of the destination from only one of the multi-homed PEs. EVPN also has mechanisms that prevent the looping of BUM traffic in an all-active multi-homed topology.

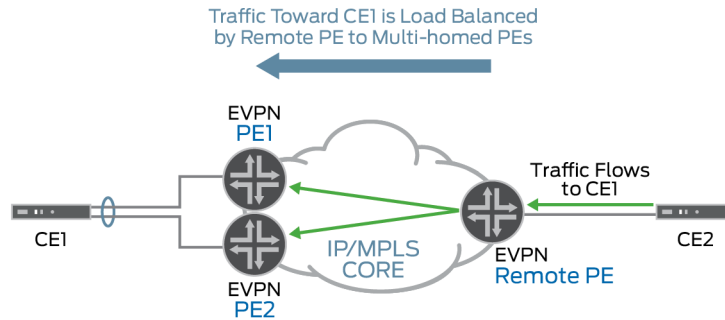


Figure 1.2 Aliasing Overview

EVPN also supports *single-active* multi-homing in which case the link(s) between a CE and only one of the PEs is active at any given time. This can be used in situations where the CE device cannot load balance traffic across all multi-homed links or the PE device cannot prevent looping of BUM traffic due to ASIC limitations. Single-active multi-homing can also make it easier to transition from existing VPLS deployments to EVPN.

## Fast Service Restoration

Multi-homing provides redundancy in the event that an access link or one of the PE routers fails. In either case, traffic flows from the CE towards the PE use the remaining active links. For traffic in the other direction, each remote PE updates its forwarding table to send traffic to the remaining active PEs, which are connected to the multi-homed Ethernet segment. EVPN provides a *Fast Convergence* mechanism so that the time it takes for the remote PEs to make this adjustment is independent of the number of MAC addresses learned by the PE.

## MAC Mobility

Data centers typically employ compute virtualization, which allows live virtual machines to be dynamically moved between hypervisors, also known as *workload migration*. EVPN's MP-BGP control plane supports *MAC Mobility*, which enables the PEs to track the movement of a VM's MAC address. Thus, the PEs always have current reachability information for the MAC address.

For example, a VM may be moved to a destination hypervisor such that it is reachable via a different PE router within the same data center or at a remote data center. After the migration is complete the VM transmits an Ethernet packet and by virtue of source MAC learning the EVPN Layer 2 forwarding table of the new PE gets updated. This PE then transmits a MAC route update to all remote PEs, which in turn update their forwarding tables. The PE that was initially local to the VM subsequently withdraws its previously advertised MAC route.

## Integration of Layer 3 Routing with Optimal Forwarding Paths

EVPN allows for the integration of Layer 3 routing to the Layer 2 domain via configuration of an IRB interface for the VLAN in the EVPN instance. The IRB interface is then placed in an IP VPN on the PE. Hosts in the EVPN use the IRB interface as their default gateway, which can route to destinations external to the data center or to other data center subnets using the IP VPN's VRF.

The IRB IP and MAC address configured on a given PE is shared with all remote PEs that are members of the EVPN, known as *Default Gateway Synchronization*. This is useful in scenarios where, for example, a VM is migrated to a remote data center. In this case the PE that is local to the VM will Proxy ARP on behalf of the learned default gateway and route the VM's outbound traffic directly towards the destination. This prevents having to backhaul traffic to the default gateway in the VM's original data center.

A PE also dynamically learns the IP addresses of the EVPN data center hosts by snooping ARP or DHCP packets. It then advertises corresponding host routes to remote EVPN PEs via MAC route updates, also called *Host MAC/IP Synchronization*. This enables a remote EVPN PE to efficiently route traffic to a given destination host using *Asymmetric IRB Forwarding*. In this implementation the Layer 2 header is rewritten by the ingress PE before sending the packet across the core, which allows the destination PE to bypass a Layer 3 lookup when forwarding the packet.

Similarly, a learned host IP address is also advertised by the PE to remote IP VPN PEs via a VPN route update. A remote IP VPN PE is then able to forward traffic to the PE closest to the data center host. Note that this method of optimized inbound routing is also compatible with MAC Mobility. For example, in the event that a VM is migrated to another data center, a PE at the destination data center learns of the new host, via ARP snooping, and transmits an VPN route update to all

members of the IP VPN. The remote IP VPN PEs update their forwarding tables and are able to forward traffic directly to a PE residing in the VM's new data center. This eliminates the need to backhaul traffic to the VM's original data center.

## Minimizing Core BUM Traffic

EVPN has several features to minimize the amount of BUM traffic in the core. First, a PE router performs Proxy ARP for the dynamically learned IP addresses of the data center hosts and default gateways. This reduces the amount of ARP traffic between data center sites. In addition, EVPN supports the use of efficient shared multicast delivery methods, such as P2MP or MP2MP LSPs, between sites.

## Data Plane Flexibility

Finally, since MAC learning is handled in the control plane this leaves EVPN with the flexibility to support different data plane encapsulation technologies between PEs. This is important because it allows EVPN to be implemented in cases where the core is not running MPLS, especially in Enterprise networks. One example of an alternative data plane encapsulation is the use of GRE tunnels. These GRE tunnels can also be secured with IPSEC if encryption is required.

**MORE** For a detailed example of EVPN DCI using GRE Tunnels please see Chapter 7 of *Day One: Building Dynamic Overlay Service-Aware Networks*, by Russell Kelly, in the *Day One* library at <http://www.juniper.net/dayone>, or on iTunes or Amazon.

In this book's test network an IP/MPLS core with RSVP-TE signaled label-switched paths (LSPs) are used to transport traffic between PEs. Given that the use of MPLS technology in the core is well understood and deployed, all inherent benefits such as fast reroute (FRR) and traffic engineering are applicable to EVPN networks as well, without any additional special configuration.

## Other Applications - EVPN with NVO

EVPN is ideally suited to be a control plane for data centers that have implemented a network virtualization overlay (NVO) solution on top of a simple IP underlay network. Within an NVO data center EVPN provides virtual Layer 2 connectivity between VMs running on different hypervisors and physical hosts. Multi-tenancy requirements of traffic and address space isolation are supported by mapping one or more VLANs to separate EVIs.

In the data plane, network overlay tunnels using VXLAN, NVGRE, or MPLS over GRE encapsulations can be used. In this case the overlay tunnel endpoint, for example a VXLAN Tunnel Endpoint (VTEP), is equivalent to a PE and runs on a hypervisor's vSwitch/vRouter or on a physical network device that supports tunnel endpoint gateway functionality.

Combining this application with EVPN DCI provides extended Layer 2 connectivity between VMs and physical hosts residing in different data centers. At each data center the overlay tunnels terminate directly into an EVI on the PE, or WAN edge, router. The EVPN then essentially “stitches” the tunnels between sites.

## Get Ready to Implement EVPN

By now you should have a better idea of how EVPN addresses many of the networking challenges presented by DCI. And hopefully your curiosity about how all of these EVPN features work has been piqued.

The next chapter reviews the test network topology, then walks you through the configuration of EVPN. Next, the book takes a deep dive into the operation of EVPN to verify that it is working properly, something you should appreciate in your own lab work. Finally, high availability tests are performed to understand the impact of link and node failures to EVPN traffic flows.

When finished you will have strong understanding of how EVPN works in addition to a working network configuration that can be used as reference. This knowledge can then be applied to helping you design, test, and troubleshoot EVPN networks with confidence.

Let's get into the lab!





# Chapter 2

## Configuring EVPN

This chapter first reviews the test network topology so that you can get oriented with the various devices and hosts. Then we'll step through the configuration of EVPN. Please refer to the *Terminology* section if you are not sure about any of the new, unfamiliar acronyms that you come across.

### The Test Network

A description of the components used to build the EVPN DCI demonstration test network is provided in the sections below. The components are separated into three groups: Core, Access, and Hosts.

#### Core

In the core, PE and P routers are various model Juniper MX routers running a pre-release version of 14.1R4. Routers PE11 and PE12 are in Data Center 1 (DC1), routers PE21 and PE22 are in Data Center 2 (DC2), and PE31 is located at a remote site. The remote site represents a generic location where there are no data center-specific devices, such as virtualized servers or storage. It could be a branch site or some other intranet site from which clients access the data centers.

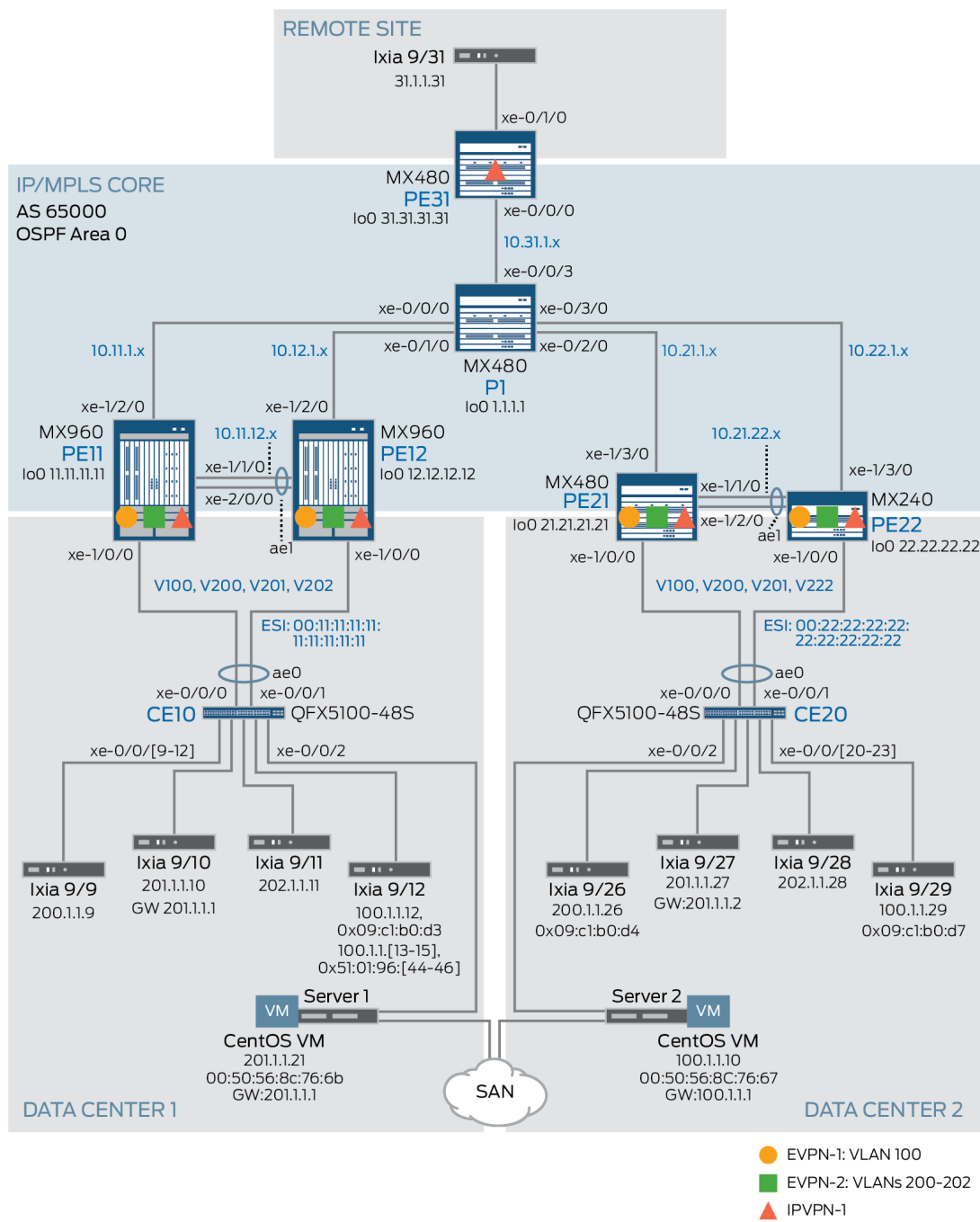


Figure 2.1 The Test Network

The IP/MPLS core is a single autonomous system. OSPF is enabled on all core interfaces to provide IP connectivity between all of the core routers. A full mesh of RSVP-TE LSPs are configured between all PEs in order to transport customer traffic between sites. PEs exchange reachability information for protocol families EVPN and IP VPN via an MP-iBGP session with the P1 route reflector. In real deployment scenarios the use of route reflectors in a redundant configuration is recommended, however for simplicity a single route reflector is used in this case.

The PEs located in the data centers are configured with two EVPN instances (EVIs). The first EVI maps to VLAN 100 and the second EVI maps to VLANs 200-202. Note that VLAN 222 in DC2 is not a typo. The local PEs will translate the VLAN ID 202 defined in the EVI to the VLAN ID 222 used in the data center.

On each PE an IRB interface is configured for each VLAN and represents the default gateway for the hosts in that VLAN. The IP and MAC address of the IRB interfaces is the same for the set of PEs in each data center. The IRB interface configuration for each VLAN may or may not be the same across data centers, as we'll see when configuring the EVIs.

Each data center PE is configured with a single IP VPN instance that includes all of the IRB interfaces. PE31 at the remote site is also a member of the IP VPN. This enables the PEs to route traffic between hosts in the EVPNs and the remote site.

Each pair of PEs in each data center is configured with a common ESI for multi-homing support. In this case, the ESI mode is set to all-active, meaning that the links connected to the CEs are both active such that traffic can be load balanced.

## Access

In each data center there is a single CE device, specifically a QFX5100-48S running Junos version 14.1X53-D10.4. The CE is configured with Layer 2 VLANs to provide connectivity between the EVI access interfaces on the PEs and the hosts. The CE is configured with a LAG bundle consisting of two uplinks, each of which terminates on a different PE. In this book's topology, all links are always active.

### IMPORTANT

If you are building your own network for testing EVPN, note that the demonstration network used in this *Day One* book can be tweaked to match your planned design or based on what hardware you have available in your lab. For example, you could eliminate the P1 router and make one of the PEs a route reflector or configure redundant route reflectors. You can configure only one of the data centers with redun-

dant PEs, or, in the access layer, you can use any device that supports LAG. You get the idea. It's recommended that you initially go through the Configuration and Verification sections of this book to get an understanding of how EVPN works. Once you're done you can go back and experiment with your own lab network. If you don't have any, or enough, equipment that's okay too; this book is written so that you can easily follow along with its lab topology.

## Hosts

A combination of emulated hosts and actual hosts is used in the test network. Each Ixia tester port emulates a single IP host in each of the four VLANs at each data center. One exception is the Ixia 9/12 port, which is in VLAN 100 and emulates four hosts. Each of the data center hosts is configured with a default gateway corresponding to the IRB interface address on the local PE's EVI VLAN. In addition, an Ixia tester port is connected to PE31 to represent a remote host or device.

### LAB NOTE

The Ixia interfaces are housed in an Ixia XM12 Chassis running IxOS 6.70.1050.14 EA-Patch2. The IxNetwork application, version 7.31.911.19 EA, is used to configure the Ixia tester interfaces as emulated hosts with the default gateway of the local PE. IxNetwork is also used to generate traffic flows and to view statistics in order to measure recovery time when performing the high availability tests in Chapter 4.

Server 1 and Server 2, in DC1 and DC2, respectively, are both Dell PowerEdge R815 servers running VMware ESXi 5.0. Each server is connected to its local data center CE device. There are two VMs, each running CentOS, that can reside on either server at any given time. The first VM is configured as a host on VLAN 100 and the second VM is configured as a host on VLAN 201. These VMs are moved between servers, using VMware vMotion, in order to verify various features of the EVPN related to MAC Mobility. Note that each server has a second connection to a common storage area network (SAN) that uses the NFS protocol. This is required in order for vMotion to work properly.

## Configuration

The focus of the configuration is on router PE11. Configuration for the other data center PEs is very similar and configuration elements from the other PEs are included here when appropriate. Reference Figure 2.1 whenever needed.

**NOTE** A cut and paste edition of this book is available for copying configurations and pasting them directly into your CLI. It is available only on this book's landing page, at <http://www.juniper.net/dayone>.

## System

EVPN requires that the MX run in `enhanced-ip` mode because it is only supported on Trio chip-based FPCs. After committing this change a reboot is required:

```
chassis {
    network-services enhanced-ip;
}
```

## Core

The core network is configured with OSPF on all interfaces for advertising and learning IP reachability, MPLS with RSVP-TE LSPs to transport data between PEs, and MP-BGP for EVPN and IP VPN signaling.

1. First configure the loopback interface based on the router number, here *11*:

```
interfaces {
    lo0 {
        unit 0 {
            family inet {
                address 11.11.11.11/32;
            }
        }
    }
}
```

2. Define the global router ID, based on the loopback interface, and the autonomous system number to be used for BGP:

```
routing-options {
    router-id 11.11.11.11;
    autonomous-system 65000;
}
```

3. Configure the core interfaces, `xe-1/2/0` and `ae1`. Assign an IP address and enable MPLS so that the interface can transmit and accept labeled packets:

```
chassis {
    aggregated-devices {
        ethernet {
            device-count 2;
        }
    }
}
```

```

    }
}

interfaces {
    xe-1/1/0 {
        gigether-options {
            802.3ad ae1;
        }
    }
    xe-1/2/0 {
        unit 0 {
            family inet {
                address 10.11.1.11/24;
            }
            family mpls;
        }
    }
    xe-2/0/0 {
        gigether-options {
            802.3ad ae1;
        }
    }
    ae1 {
        aggregated-ether-options {
            lacp {
                active;
            }
        }
        unit 0 {
            family inet {
                address 10.11.12.11/24;
            }
            family mpls;
        }
    }
}

```

4. Next, enable OSPF, MPLS, and RSVP protocols on the loopback and core interfaces. Note that `traffic-engineering` is enabled under OSPF, which creates a traffic engineering database (TED). The TED is used to determine the path for each LSP that is subsequently signaled and established using RSVP-TE:

```

protocols {
    rsvp {
        interface xe-1/2/0.0;
        interface lo0.0;
        interface ae1.0;
    }
    mpls {
        interface ae1.0;
        interface xe-1/2/0.0;
    }
    ospf {

```

```

    traffic-engineering;
    area 0.0.0.0 {
        interface ae1.0;
        interface xe-1/2/0.0;
        interface lo0.0;
    }
}

```

5. Create the LSPs to each of the other PEs. These LSPs will be used by both EVPN and IP VPN services:

```

protocols {
    mpls {
        label-switched-path from-11-to-12 {
            from 11.11.11.11;
            to 12.12.12.12;
        }
        label-switched-path from-11-to-21 {
            from 11.11.11.11;
            to 21.21.21.21;
        }
        label-switched-path from-11-to-22 {
            from 11.11.11.11;
            to 22.22.22.22;
        }
        label-switched-path from-11-to-31 {
            from 11.11.11.11;
            to 31.31.31.31;
        }
    }
}

```

6. Finally, configure the MP-BGP session to P1 whose loopback address is 1.1.1.1. It's important to explicitly set the `local-address` because we want to establish the sessions between loopback addresses. By default the IP address of the interface closest to the neighbor is used.

The protocol families EVPN and IP VPN are configured corresponding to the service instances configured on the PE. Also, BFD is enabled for faster failure detection in the event that the router fails (see *Chapter 4 High Availability Tests - Node Failure* test case):

```

protocols {
    bgp {
        group Internal {
            type internal;
            family inet-vpn {
                any;
            }
            family evpn {
                signaling;
            }
            neighbor 1.1.1.1 {
                local-address 11.11.11.11;
            }
        }
    }
}

```

```

        bfd-liveness-detection {
            minimum-interval 200;
            multiplier 3;
        }
    }
}

```

## Access

PE11 has a single access interface connected to CE10. The interface carries the multiple VLANs that map to the different EVPN instances (EVI). In this case, a logical interface is configured for each EVI. Unit 100 contains a single VLAN 100 that maps to instance EVPN-1 and unit 200 contains three VLANs that map to instance EVPN-2.

An ESI is required for EVPN multi-homing, a 10 octet value that must be unique across the entire network. According to the EVPN standard, the first octet represents the *Type* and the remaining 9 octets are the ESI value. Currently Junos allows all 10 octets to be configured to any value.

In this lab network the first byte of the ESI is set to 00, which means the remaining 9 octets of the ESI value are set statically. The same exact ESI value must be configured on PE12, the multi-homing peer PE. If a CE has only a single connection to a PE then the ESI must be 0 which is the default value.

The multi-homing mode of *all-active* is configured indicating that both multi-homed links between the CE and the PEs are always active. This allows traffic from the CE and remote PEs to be load balanced between the two multi-homed PEs.

**NOTE** *Single-active* mode is also supported where only one multi-homed link is active at any given time.

Note that the access interface is configured as a LAG with a single link member. The reason is that it is desirable to enable LACP at the access layer to control initialization of the interface. Used in conjunction with the hold-up timer, which is set at the physical interface level, this configuration minimizes packet loss in the event of a link or node recovery. We'll see these mechanisms in action in *Chapter 4, High Availability Tests*.

In order for the LAG to work properly the system-id must be set to the same value on both multi-homed PEs. This tricks the CE into thinking



that it is connected to a single device and ensures that the LACP negotiation is successful.

The important point here is that the PE11 and PE12 routers identify each multi-homed link based on the ESI value. The LAG configuration is completely independent of EVPN multi-homing, and it is not required when there is a single link between the PE and CE. For example, if the ESI and VLANs were configured on the xe-1/0/0 interface without any LAG, the EVPN multi-homing solution would still work. The only purpose of the LAG configuration is to improve the resiliency in the access network when the link comes up.

In this lab topology there is a single link between each PE and CE; however, configurations consisting of multiple links bundled into a LAG are also supported. In these cases it is required to configure a LAG between each PE and CE including a common, static System ID on each of the multi-homed PEs. If the CE supports LACP, then it should be enabled on both ends of the link as well:

```
interfaces {
  xe-1/0/0 {
    hold-time up 180000 down 0;
    gigether-options {
      802.3ad ae0;
    }
  }

  ae0 {
    flexible-vlan-tagging;
    encapsulation flexible-ethernet-services;
    esi {
      00:11:11:11:11:11:11:11:11;
      all-active;
    }
    aggregated-ether-options {
      lacp {
        system-id 00:00:00:00:00:01;
      }
    }
    unit 100 {
      encapsulation vlan-bridge;
      vlan-id 100;
      family bridge;
    }
    unit 200 {
      family bridge {
        interface-mode trunk;
        vlan-id-list [ 200 201 202 ];
      }
    }
  }
}
```

The CE10 configuration below shows that the multi-homed links are configured as a LAG bundle with LACP enabled.

```

interfaces {
  xe-0/0/0 {
    hold-time up 180000 down 0;
    ether-options {
      802.3ad ae0;
    }
  }
  xe-0/0/1 {
    hold-time up 180000 down 0;
    ether-options {
      802.3ad ae0;
    }
  }
  ae0 {
    aggregated-ether-options {
      lacp {
        active;
      }
    }
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members all;
        }
      }
    }
  }
}

```

## Services

Junos for the MX Series currently supports two types of EVPN services: VLAN-based and a VLAN-aware bundle. The VLAN-based service supports a single VLAN, which maps to a single bridge domain. Note that the VLAN-aware bundle service is the most typical and preferred deployment option for DCI as it provides support for multiple tenants with independent VLAN and IP subnet space.

Both of these services support VLAN translation, meaning that different VLAN Identifiers can be used at different sites. The PE router is responsible for performing translation between the local VLAN ID and the Ethernet Tag Identifier, or VLAN ID used by the service.

In order to route between VLANs, each VLAN is configured with an IRB interface that acts as a default gateway for the hosts in the EVI. The IRB interfaces are placed into a common IP VPN so that inter-VLAN routing can occur. The IP VPN also allows routing to and from other non-data center networks.

In this lab network a single IP VPN service is configured. In practice you may put different IRB interfaces into different IP VPN VRFs or the inet.0 table in order to provide separation of Layer 3 traffic. You may also have some EVPNs that do not require integrated Layer 3 functionality, for example, in cases where a firewall is used as the default gateway to enforce security policy.

A summary of the configured services is listed in Table 2.1, with details on each service following the table. Note that the various configurations will be used to verify the features of the EVPN DCI solution.

**Table 2.1**      **EVPN and IPVPN Services**

Service Name	Service Type	VLAN	Configuration Notes
EVPN-1	EVPN - VLAN Based	100	PEs in DC1 and DC2 configured with same Default Gateway.
EVPN-2	EVPN - VLAN Aware Bundle	200	PEs in DC1 and DC2 configured with same Default Gateway.
		201	PEs in DC1 and DC2 configured with different Default Gateways.
		202	PEs in DC1 and DC2 configured with same Default Gateway. PEs in DC2 perform VLAN translation between normalized service VLAN ID 202 and local VLAN ID 222 in DC2.
IPVPN-1	IP VPN - VRF		Configured on all PEs in DC1 and DC2 and PE31 at the remote site.

### EVPN-1

EVPN-1 is a VLAN-based EVPN service, therefore it is configured with `instance-type evpn`. The Route Distinguisher (RD) must be set to a network-wide unique value to prevent overlapping routes between different EVIs. It is recommended to use a Type 1 RD where the value field is comprised of an IP address, typically the loopback address, of the PE followed by a number unique to the PE. In this case the RD is set under the routing instance, but as an alternative the `routing-options route-distinguisher-id <ip-address>` setting can be used to automatically assign non-conflicting RDs.

The `vrf-target`, or Route Target Extended Community, is used to control the distribution of MP-BGP routes into the PE's EVI route tables. It includes the AS number followed by an identifier that is unique to the EVPN service. In this case the Route Target is set to the same value on all data center PEs for this specific EVI.

VLAN 100 on access interface `ae0.100` is the single VLAN mapped to the service. It is configured with an IRB interface, `irb.100`. The IRB interface is configured with the default gateway IP address for the VLAN and a static MAC address. In this case all of the PEs in both DC1 and DC2 are configured with the same IP and MAC addresses. Therefore, the `evpn default-gateway do-not-advertise` setting instructs the PE to not advertise the MAC/IP binding corresponding to the IRB as Default Gateway Synchronization is not required for this EVI.

**BEST PRACTICE** Configure the same IP and MAC address on all PEs for a given EVPN VLAN to simplify the configuration, reduce control plane overhead, and minimize the recovery time in the event a PE node fails:

```
routing-instances {
  EVPN-1 {
    instance-type evpn;
    vlan-id 100;
    interface ae0.100;
    routing-interface irb.100;
    route-distinguisher 11.11.11.11:1;
    vrf-target target:65000:1;
    protocols {
      evpn {
        default-gateway do-not-advertise;
      }
    }
  }
}

interfaces {
  irb {
    unit 100 {
      family inet {
        address 100.1.1.1/24;
      }
      mac 00:00:00:01:01:01;
    }
  }
}
```

## EVPN-2

EVPN-2 is a VLAN-aware bundle EVPN service, therefore it is configured with `instance-type virtual-switch`. The Route Distinguisher is set to a unique value and the Route Target is set to the same value on all data center PEs in this topology corresponding to this EVI.

VLANs 200 to 202 are configured on access interface `ae0.200`, which is mapped to the service. Note that the three VLANs, or bridge-domains, are configured within the routing instance. Each VLAN is configured with a normalizing VLAN ID and an IRB interface. The `extended-vlan-list` indicates that all VLANs are to be extended across the core.

## BEST PRACTICE

Configure the VLAN-aware bundle service even if the EVI is mapped to a single VLAN. This service provides the most flexibility and allows for an easier transition in cases where changes to the service, such as adding more VLANs to the EVI, need to be made in the future.

For VLAN 200, the `irb.200` interface is configured with the same default gateway IP address and MAC address on all PEs in both DC1 and DC2, which is similar to the configuration for the preceding EVPN-1 VLAN 100.

For VLAN 201, the `irb.201` interface is configured with a different default gateway IP address and MAC address in each data center. In DC1, the default gateway on both PEs is configured with IP address 201.1.1.1 and MAC address 0xc9:01:01:01. In DC2, the default gateway on both PEs is configured with IP address 201.1.1.2 and MAC address 0xc9:01:01:02. The Ixia tester ports representing hosts in VLAN 201 are configured with the local data center default gateway as seen in Figure 2.1. Also, the VM host in VLAN 201 is initially in DC1 and is configured with a default gateway of 201.1.1.1. In *Chapter 3: Verification*, this VM will be moved to DC2 to verify that the PE routers in DC2 will route traffic received from the VM:

```
routing-instances {
  EVPN-2 {
    instance-type virtual-switch;
    interface ae0.200;
    route-distinguisher 11.11.11.11:2;
    vrf-target target:65000:2;
    protocols {
      evpn {
        extended-vlan-list 200-202;
        default-gateway advertise;
      }
    }
  }
  bridge-domains {
    V200 {
```

```

        vlan-id 200;
        routing-interface irb.200;
    }
    V201 {
        vlan-id 201;
        routing-interface irb.201;
    }
    V202 {
        vlan-id 202;
        routing-interface irb.202;
    }
}
}
}

interfaces {
    irb {
        unit 200 {
            family inet {
                address 200.1.1.1/24;
            }
            mac 00:00:c8:01:01:01;
        }
        unit 201 {
            family inet {
                address 201.1.1.1/24;
            }
            mac 00:00:c9:01:01:01;
        }
        unit 202 {
            family inet {
                address 202.1.1.1/24;
            }
            mac 00:00:ca:01:01:01;
        }
    }
}
}

```

For VLAN 202, the irb.202 interface is configured with the same default gateway IP address and MAC address on all PEs in both DC1 and DC2. However, the VLAN ID used in DC1 is 202 while the VLAN ID used in DC2 is 222 in order to demonstrate VLAN translation. In this case the PEs in DC2 translate the Ethernet Tag Identifier, or VLAN ID, defined in the EVI to the local VLAN ID that is understood by CE20. This is accomplished using the `vlan-rewrite` parameter under the access interface configuration on PE21 and PE22:

```

interfaces {
    ae0 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        esi {

```

```

    00:22:22:22:22:22:22:22:22;
    all-active;
}
aggregated-ether-options {
    lacp {
        system-id 00:00:00:00:00:02;
    }
}
unit 100 {
    encapsulation vlan-bridge;
    vlan-id 100;
    family bridge;
}
unit 200 {
    family bridge {
        interface-mode trunk;
        vlan-id-list [ 200 201 202 ];
        vlan-rewrite {
            translate 222 202;
        }
    }
}
}
}
}
}

```

### IPVPN-1

A single IP VPN instance named IPVPN-1 with instance-type `vrf` is configured on each PE router. This service instance contains the IRB interfaces of all the EVPN VLANs on the data center PEs, that populates the local VRF with the IP subnets corresponding to the EVPN VLANs.

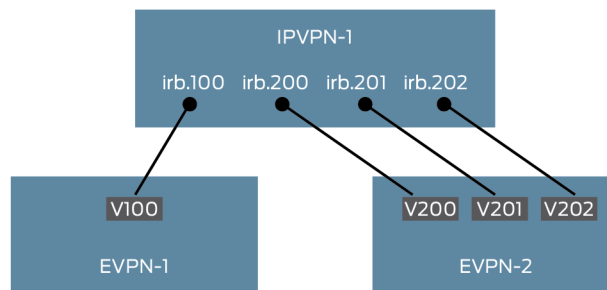


Figure 2.2 EVPN IRB Placement into IP VPN

More importantly, whenever the PE learns a new host IP address in an EVPN VLAN configured with an IRB, it automatically adds a corresponding host route to the VRF. This host route gets advertised to all

remote PEs and allows traffic from other EVPN VLANs or remote sites destined to the host to be optimally forwarded. This is explained in more detail in *Chapter 3: Verification - Layer 3 Operations*:

```
routing-instances {
  IPVPN-1 {
    instance-type vrf;
    interface irb.100;
    interface irb.200;
    interface irb.201;
    interface irb.202;
    route-distinguisher 11.11.11.11:111;
    vrf-import IpVpnDiscardEvpnSubnets;
    vrf-export IpVpnAddCommunities;
    vrf-table-label;
  }
}
```

The IPVPN-1 routing instance Route Distinguisher is set to a network-wide unique value on each of the PEs in the topology. On the data center PEs, a vrf-export policy is configured to set community COMM-EVPN in addition to the required VRF Route Target community, COMM-IPVPN-1 for the advertised routes. This effectively tags IP VPN routes that correspond to any of the EVPN subnets and hosts. A data center PE that receives IP VPN routes with community COMM-EVPN will reject them, via a vrf-import policy, because these same routes are learned via EVPN MAC/IP Advertisement routes. Discarding the redundant IP VPN routes reduces the number of VRF table entries. This is explained in more detail in *Chapter 3: Verification - Layer 3 Operations*:

```
policy-options {
  prefix-list PL-EVPN {
    100.1.1.0/24;
    200.1.1.0/24;
    201.1.1.0/24;
    202.1.1.0/24;
  }
  policy-statement IpVpnAddCommunities {
    term 10 {
      from {
        prefix-list-filter PL-EVPN orlonger;
      }
      then {
        community add COMM-EVPN;
        community add COMM-IPVPN-1;
        accept;
      }
    }
    term 100 {
      then accept;
    }
  }
}
```



```

}
policy-statement IpVpnDiscardEvpnSubnets {
    term 10 {
        from community COMM-EVPN;
        then reject;
    }
    term 100 {
        from community COMM-IPVPN-1;
        then accept;
    }
}
community COMM-EVPN members 65000:1234;
community COMM-IPVPN-1 members target:65000:101;
}

```

#### PE31 Configuration

On PE31, the local xe-0/1/0.0 interface, corresponding to IP subnet 31.1.1/24, is included in IPVPN-1. All of the IP VPN routes received by PE31 originating from the data center PEs contain both COMM-IPVPN-1 and COMM-EVPN communities. The IPVPN-1 routing instance on PE31 places all of these IP VPN routes into its VRF since it is configured to accept all routes with Route Target community target:65000:101. The COMM-EVPN community 65000:1234 is simply ignored:

```

routing-instances {
    IPVPN-1 {
        instance-type vrf;
        interface xe-0/1/0.0;
        route-distinguisher 31.31.31.31:101;
        vrf-target target:65000:101;
        vrf-table-label;
    }
}

```

PE31 may receive IP VPN routes for a given data center host from both multi-homed PEs in that data center. In order to load balance traffic to both data center PEs, BGP multipath is enabled on PE31 along with a load-balance per-packet policy. This enables Junos to install multiple next hops for a given destination learned via BGP in the Packet Forwarding Engine (PFE):

```

bgp {
    group Internal {
        type internal;
        family inet-vpn {
            any;
        }
        multipath;
        neighbor 1.1.1.1 {
            local-address 31.31.31.31;
            bfd-liveness-detection {

```

```

        minimum-interval 200;
        multiplier 3;
    }
}

policy-options {
    policy-statement lb {
        then {
            load-balance per-packet;
        }
    }
}

routing-options {
    router-id 31.31.31.31;
    autonomous-system 65000;
    forwarding-table {
        export lb;
    }
}

```

## Default Configuration

Starting in Junos Release 14.1R4, the load balancing and chained composite next hop features required for EVPN are automatically configured.

Aliasing is an EVPN feature that allows for load balancing of traffic flows towards a pair of PEs in an all-active multi-homing configuration. For example, when PE11 sends traffic to a destination in DC2, it can load balance the traffic to PE21 and PE22, which are both connected to the same Ethernet Segment, or CE device. Aliasing is applied to Layer 2 and Layer 3 traffic flows between EVPN VLANs and is explored in detail in *Chapter 3: Verification*.

Chained composite next hops are required to efficiently route traffic between hosts or devices on different EVPN VLANs, connected to different PEs, using a process called asymmetric IRB forwarding. The chained composite next hop essentially allows the EVPN PE to perform multiple next hop actions on a packet. In this case, the ingress PE rewrites the Ethernet header of the original packet and then pushes the appropriate MPLS labels before forwarding it. This enables the destination PE to bypass a VRF table lookup on egress. In *Chapter 3: Verification*, the *Layer 3 Operations – Inter-VLAN Routing* section has more details on this feature.

The default configuration can be viewed using the following CLI commands:

```
cse@PE11> show configuration routing-options forwarding table | display inheritance defaults
```

```
##
## 'evpn-pplb' was inherited from group 'junos-defaults'
##
export evpn-pplb;
##
## 'chained-composite-next-hop' was inherited from group 'junos-defaults'
##
chained-composite-next-hop {
  ##
  ## 'ingress' was inherited from group 'junos-defaults'
  ##
  ingress {
    ##
    ## 'evpn' was inherited from group 'junos-defaults'
    ##
    evpn;
  }
}
```

```
cse@PE11> show configuration policy-options | display inheritance defaults
```

```
##
## 'evpn-pplb' was inherited from group 'junos-defaults'
##
policy-statement evpn-pplb {
  ##
  ## 'from' was inherited from group 'junos-defaults'
  ##
  ##
  ## 'evpn' was inherited from group 'junos-defaults'
  ##
  from protocol evpn;
  ##
  ## 'then' was inherited from group 'junos-defaults'
  ##
  then {
    ##
    ## 'load-balance' was inherited from group 'junos-defaults'
    ## 'per-packet' was inherited from group 'junos-defaults'
    ##
    load-balance per-packet;
  }
}
```



# Chapter 3

## Verification

Now that EVPN has been configured in the test topology it is time to verify its operation. We'll start in the IP/MPLS core to ensure that the protocols that provide the foundation of the services are functioning as expected. Then we'll make sure that the EVPN and IP VPN services are in the correct state and ready to forward traffic. From the CLI we will delve into the inner workings of EVPN multi-homing, come to understand how the EVPN and IP VPN forwarding tables are populated, and learn how unicast and multi-cast traffic is forwarded. Along the way traffic flows will be generated to demonstrate some of the key features such as aliasing and Layer 3 traffic path optimization. The goal is to gain a better understanding of the underlying mechanisms that enable the many beneficial features of EVPN.

## Core

The operation in the core is similar to that of other VPN services. So let's quickly verify that the various IP control protocols including OSPF, BGP, and RSVP-TE are running properly.

First confirm that the OSPF adjacencies are Full because a problem here will prevent the other protocols from working. Here is the output from PE11:

```
cse@PE11> show ospf neighbor
```

Address	Interface	State	ID	Pri	Dead
10.11.12.12	ae1.0	Full	12.12.12.12	128	39
10.11.1.1	xe-1/2/0.0	Full	1.1.1.1	128	31

Next, check that the state of the MP-BGP session to P1 is Established. If the services are configured correctly you should see primary route tables for IP VPN and EVPN, `bgp.13vpn.0` and `bgp.evpn.0`, respectively, that contain all of the routes received from the route reflector. In addition, the secondary route tables `IPVPN-1.inet.0`, `EVPN-1.evpn.0`, `EVPN-2.evpn.0`, and `__default__evpn__.evpn.0` should be present. Also verify that the BGP based BFD session to P1 is Up:

```
cse@PE11> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table Tot Paths Act Paths Suppressed History Damp State Pending
bgp.13vpn.0
          21         21          0          0          0          0
bgp.13vpn.2
          0          0          0          0          0          0
bgp.evpn.0
          46         46          0          0          0          0
Peer      AS      InPkt    OutPkt    OutQ    Flaps Last Up/
Dwn State|#Active/Received/Accepted/Damped...
1.1.1.1    65000    3187     3066      0        0    22:47:31 Estab1
  bgp.13vpn.0: 21/21/21/0
  bgp.13vpn.2: 0/0/0/0
  bgp.evpn.0: 46/46/46/0
  IPVPN-1.inet.0: 1/21/21/0
  EVPN-1.evpn.0: 14/14/14/0
  EVPN-2.evpn.0: 34/34/34/0
  __default__evpn__.evpn.0: 1/1/1/0
```

```
cse@PE11> show bfd session

Address          State      Interface      Detect   Transmit
1.1.1.1          Up         Up              Time    Interval Multiplier
0.600           0.200         3
```

```
1 sessions, 1 clients
Cumulative transmit rate 5.0 pps, cumulative receive rate 5.0 pps
```

Confirm that the LSPs from PE11 to all remote PEs are Up. Similarly, the LSPs from remote PEs that terminate on PE11 should be Up:

```
cse@PE11> show mpls lsp
Ingress LSP: 4 sessions
To      From      State Rt P    ActivePath      LSPname
12.12.12.12 11.11.11.11 Up    0 *      from-11-to-12
21.21.21.21 11.11.11.11 Up    0 *      from-11-to-21
22.22.22.22 11.11.11.11 Up    0 *      from-11-to-22
31.31.31.31 11.11.11.11 Up    0 *      from-11-to-31
Total 4 displayed, Up 4, Down 0

Egress LSP: 4 sessions
To      From      State Rt Style Labelin Labelout LSPname
11.11.11.11 22.22.22.22 Up    0 1 FF      3      - from-22-to-11
11.11.11.11 12.12.12.12 Up    0 1 FF      3      - from-12-to-11
11.11.11.11 31.31.31.31 Up    0 1 FF      3      - from-31-to-11
```

```
11.11.11.11      21.21.21.21      Up      0  1 FF      3      - from-21-to-11
Total 4 displayed, Up 4, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

## Access

Let’s quickly verify that the interface between PE11 and CE10 is up and that LACP has successfully been negotiated. This is important because an issue here prevents the EVIs from initializing:

```
cse@PE11> show interfaces terse | match ae0
xe-1/0/0.100      up    up    aenet  --> ae0.100
xe-1/0/0.200      up    up    aenet  --> ae0.200
xe-1/0/0.32767    up    up    aenet  --> ae0.32767
ae0               up    up
ae0.100           up    up    bridge
ae0.200           up    up    bridge
ae0.32767         up    up    multiservice

cse@PE11> show lacp interfaces ae0
Aggregated interface: ae0
  LACP state:      Role  Exp  Def  Dist  Col  Syn  Aggr  Timeout  Activity
    xe-1/0/0      Actor No   No   Yes   Yes Yes   Yes   Fast   Passive
    xe-1/0/0      Partner No   No   Yes   Yes Yes   Yes   Fast   Active
  LACP protocol:      Receive State  Transmit State      Mux State
    xe-1/0/0              Current  Fast periodic Collecting distributing
```

## Multi-homing

Multi-homing is an important requirement of data center network designs because it provides resiliency in the event of a node or link failure to ensure that critical applications stay up and running. It also enables the efficient use of bandwidth by load balancing bi-directional traffic on all links between the PEs and CE and optimally forwarding BUM traffic to prevent Layer 2 broadcast storms. The mechanisms that enable these attributes of EVPN are explored in the sections below.

### Discovering Ethernet Segments

As discussed in Chapter 2, multi-homing is configured by setting the ESI value on the access interface to the same value on both PEs. This triggers the advertisement of an MP-BGP Ethernet Segment route by each of the multi-homed PEs that allows them to automatically discover each other.

The ES route, NLRI Route Type 4, contains the following key fields:

- Originator Router's IP Address – loopback address of the advertising PE.
- ESI – network-wide unique 10-byte Ethernet Segment Identifier.
- ES-Import Route Target Extended Community - automatically derived from the ESI.

The EVPN standard states that an ES route must only be accepted by a PE that is connected to the same ES. Therefore, the PE receiving the route evaluates the ES-Import community to determine whether or not it was sent by a multi-homed peer PE connected to the same ES.

Note that the ES-Import community only encodes six of the nine ESI value bytes. Although there is a chance that two different ESI values might map to the same ES-Import community, this first level of filtering still greatly reduces the number of individual routes that the PE needs to process. When the PE subsequently performs the Designated Forwarder election, it matches on the full ESI value (refer to next section for details).

### Lab Example – ES Route

Let's now turn to the lab topology for an example. The ES route received by PE11 from PE12 is displayed below. In this case PE11 accepts the ES route because the `es-import-target` value 11-11-11-11-11-11 corresponds to PE11's configured ESI. Similarly, PE12 accepts PE11's ES route advertisement. The PEs in Data Center 2 discard these routes because the ES-Import community does not correspond to their locally configured ESI. Also note that in this lab topology there are a pair of multi-homed PEs in each data center, however configurations consisting of more than two multi-homed PEs are also supported:

```
cse@PE11> show route table bgp.evpn.0 detail | find "4:\1"
4:12.12.12.12::1111111111111111:12.12.12.12/304 (1 entry, 0 announced)
    *BGP      Preference: 170/-101
              Route Distinguisher: 12.12.12.12:0
              Next hop type: Indirect
              Address: 0x95c606c
              Next-hop reference count: 25
              Source: 1.1.1.1
              Protocol next hop: 12.12.12.12
              Indirect next hop: 0x2 no-forward INH Session ID: 0x0
              State: <Active Int Ext>
              Local AS: 65000 Peer AS: 65000
              Age: 20:37:39 Metric2: 1
              Validation State: unverified
              Task: BGP_65000.1.1.1.1+179
              AS path: I (Originator)
```



```

Cluster list: 1.1.1.1
Originator ID: 12.12.12.12
Communities: es-import-target:11-11-11-11-11
Import Accepted
Localpref: 100
Router ID: 1.1.1.1
Secondary Tables: __default_evpn__.evpn.0

```

All EVPN NLRI Type 4 routes are also stored in the secondary \_\_default\_evpn\_\_.evpn.0 table since they do not contain a Route Target community that corresponds to any specific EVI. The output below shows the Type 4 ES route that originates from the local PE as well as the received and accepted ES route from PE12:

```

cse@PE11> show route table __default_evpn__.evpn.0 | find 4:
4:11.11.11.11:0::1111111111111111:11.11.11.11/304
    *[EVPN/170] 04:40:23
    Indirect
4:12.12.12.12:0::1111111111111111:12.12.12.12/304
    *[BGP/170] 04:40:04, localpref 100, from 1.1.1.1
    AS path: I, validation-state: unverified
    > to 10.11.12.12 via ae1.0, label-switched-path from-11-to-12

```

## Designated Forwarder

Once a set of multi-homed peers have discovered each other, one of PEs is elected as the Designated Forwarder (DF) for the ES. The DF is responsible for transmitting broadcast, unknown unicast, and multi-cast (BUM) traffic from the core to the CE. The non-DF, or Backup Forwarder, PE drops BUM traffic received from the core destined to the CE.

### Lab Example - DF

From the lab topology, the EVPN-1 instance information on PE11 shows that PE11 is the Designated forwarder and PE12 is the Backup forwarder:

```

cse@PE11> show evpn instance EVPN-1 esi 00:11:11:11:11:11:11:11:11:11 extensive
Instance: EVPN-1
<snip>
  Number of ethernet segments: 2
  ESI: 00:11:11:11:11:11:11:11:11:11
  Status: Resolved by IFL ae0.100
  Local interface: ae0.100, Status: Up/Forwarding
  Number of remote PEs connected: 1
    Remote PE      MAC label  Aliasing label  Mode
    12.12.12.12    300688     300688          all-active
Designated forwarder: 11.11.11.11
Backup forwarder: 12.12.12.12
  Advertised MAC label: 300976

```

Advertised aliasing label: 300976  
 Advertised split horizon label: 299984

The DF election is performed at the granularity of per ESI per EVI. This facilitates the load balancing of BUM traffic amongst the PEs, a feature also known as *Service Carving*. Therefore, for EVI EVPN-2 a separate DF election takes place, and PE11 is again the DF:

```
cse@PE11> show evpn instance EVPN-2 esi 00:11:11:11:11:11:11:11:11 extensive
Instance: EVPN-2
<snip>
Number of ethernet segments: 2
ESI: 00:11:11:11:11:11:11:11:11
Status: Resolved by IFL ae0.200
Local interface: ae0.200, Status: Up/Forwarding
Number of remote PEs connected: 1
  Remote PE      MAC label  Aliasing label  Mode
  12.12.12.12    300144      300144          all-active
Designated forwarder: 11.11.11.11
Backup forwarder: 12.12.12.12
Advertised MAC label: 300080
Advertised aliasing label: 300080
Advertised split horizon label: 299984
```

According to the EVPN standard, each of the multi-homed PEs independently executes the same algorithm to determine which one is the DF. First, all of the PEs are sorted into a numerically ascending ordered list based on their Originator Router's IP Address field in the ES route, the loopback address in this case. Next, each PE is assigned an index value starting at 0. For example, in our lab PE11 is assigned 0 and PE12 is assigned 1.

Then the result of the formula ( $V \bmod N$ ), where  $V$  is the VLAN ID and  $N$  is the number of multi-homed PE nodes, is used to determine which PE in the list is the DF. If there are multiple VLANs associated with the EVI then the lowest value is used. Therefore, for the EVIs in this lab configuration the values 100 and 200 are used for  $V$  and the result of the DF election formula is 0, or PE11, for both instances.

From the CLI the details of the logical access interface also indicate the winner of the DF election. In the output below we see that the logical access interfaces for EVPN-1 and EVPN-2 on PE11 are in the Forwarding state:

```
cse@PE11> show interfaces ae0.100 detail | find EVPN
EVPN multi-homed status: Forwarding, EVPN multi-homed ESI Split Horizon
Label: 299984
Flags: Is-Primary

cse@PE11> show interfaces ae0.200 detail | find EVPN
EVPN multi-homed status: Forwarding, EVPN multi-homed ESI Split Horizon
Label: 299984
```

Flags: Is-Primary, Trunk-Mode

The multi-homed status of the corresponding interfaces on PE12, the non-DF, are Blocking BUM Traffic to ESI:

```
cse@PE12> show interfaces ae0.100 detail | find EVPN
```

```
  EVPN multi-homed status: Blocking BUM Traffic to ESI, EVPN multi-homed ESI Split  
Horizon Label: 299888
```

```
  Flags: Is-Primary
```

```
cse@PE12> show interfaces ae0.200 detail | find EVPN
```

```
  EVPN multi-homed status: Blocking BUM Traffic to ESI, EVPN multi-homed ESI Split  
Horizon Label: 299888
```

```
  Flags: Is-Primary, Trunk-Mode
```

## Auto-Discovery per ESI and per EVI

In a multi-homed configuration, each PE router advertises two types of Auto-Discovery routes to all other PEs via MP-BGP. These advertisements are referred to as Auto-Discovery per ESI and Auto-Discovery per EVI.

### Auto-Discovery per ESI

The Auto-Discovery per ESI route is used for fast convergence and for preventing the looping of BUM traffic. It is a mandatory route that is advertised by both multi-homed PEs connected to the ES. The advertised route includes the following data:

- A list of Route Targets corresponding to the EVPN instances associated with the ESI
- The ESI value
- ESI Label Extended Community – contains an MPLS Split Horizon label and the multi-homing mode, single-active or all-active

When a remote PE router that is configured with matching route targets, or EVPN instances, receives this advertisement, it has a view of the multi-homing connectivity of the advertising PEs. One benefit here is for fast convergence, also known as *MAC Mass Withdraw*. In the event a multi-homed PE loses its local link towards the CE, it withdraws this route. This signals to the remote PEs to either invalidate or adjust the next hop of all MAC addresses that correspond to the advertising PE's failed Ethernet Segment. This is more efficient than requiring the PE to withdraw each individual MAC address in which case the convergence time would be dependent on the scale, or total number, of MAC addresses.

The MPLS Split Horizon label, also called the ESI MPLS label, is used to prevent looping of multi-destination traffic amongst multi-homed PE peers, also known as *Split Horizon Filtering*. In an all-active multi-homing topology, when a non-DF PE forwards a BUM packet to its peer DF PE, it first pushes this received label onto the packet. Then it pushes the Inclusive Multicast label (see the *Inclusive Multicast* section below) followed by the transport label to reach the loopback of the destination peer PE.

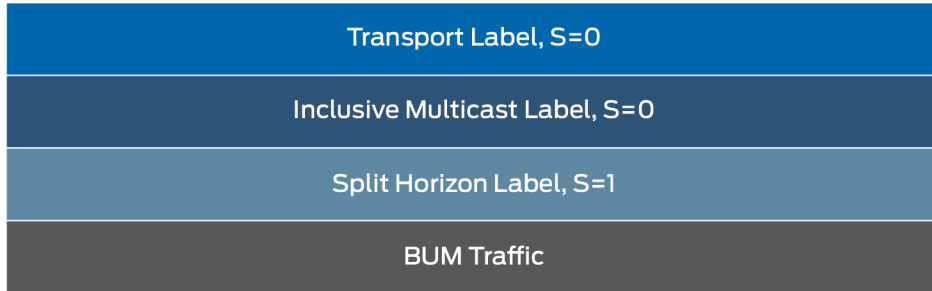


Figure 3.1 MPLS Encapsulation of BUM Traffic by non-DF PE

When the DF PE receives and inspects the MPLS labels in the packet, it recognizes the Split Horizon label it previously advertised and does not forward the packet back to the CE.

#### Auto-Discovery per EVI

The Auto-Discovery per EVI route is an optional route that is advertised by the multi-homed PEs. In an all-active multi-homed scenario this route is used to implement the EVPN aliasing, or load balancing, feature that has been mentioned previously. For example, one of the multi-homed PEs could be advertising all, or a majority of the MAC addresses learned from the CE, to the remote PEs. The remote PEs in turn would only send traffic to the advertising PE. Aliasing allows the other multi-homed peer PE, which may not have learned/advertised any MAC addresses, to also receive traffic from remote PEs destined to the common ES.

In single-active multi-homed mode this route is used to implement a similar *Backup-path* feature. In this case, a remote PE sends traffic to the multi-homed PE that is the DF and installs a backup forwarding entry pointing to the non-DF PE.

The Auto-Discovery per EVI route includes the following key parameters:

- The Route Target corresponding to the EVI

- The ESI value(s) connected to the EVI
- The Aliasing label

When a PE router learns a new MAC address, it transmits an EVPN MAC Advertisement route that includes the MAC address, an MPLS service label, and the ESI it was learned on to the remote PEs. A given remote PE correlates this ESI with the ESI values in the two Auto-Discovery routes and determines the set of multi-homed PEs that it can transmit to when forwarding packets to the MAC address.

When the remote PE sends a packet destined to the MAC address to the PE that sent the MAC Advertisement route, it uses the service label. When the remote PE sends a packet destined to the advertising PEs multi-homed peer PE, which is connected to the same ES, it uses the aliasing label. This assumes that the remote PE has not received an equivalent MAC Advertisement route from the multi-homed partner PE. As we'll see in the sections below, aliasing applies to forwarding both Layer 2 and Layer 3 traffic.

### Lab Example - Auto-Discovery

In our test network, PE11 receives two Auto-Discovery routes from each of the three other PEs corresponding to EVI EVPN-1. These routes are EVPN NLRI Route Type 1:

```
cse@PE11> show route table EVPN-1.evpn.0
```

```
EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
<snip>
```

```
1:12.12.12.12:0::1111111111111111::FFFF:FFFF/304
```

```
    *[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
```

```
    AS path: I, validation-state: unverified
```

```
    > to 10.11.12.12 via ae1.0, label-switched-path from-11-to-12
```

```
1:12.12.12.12:1::1111111111111111::0/304
```

```
    *[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
```

```
    AS path: I, validation-state: unverified
```

```
    > to 10.11.12.12 via ae1.0, label-switched-path from-11-to-12
```

```
1:21.21.21.21:0::2222222222222222::FFFF:FFFF/304
```

```
    *[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
```

```
    AS path: I, validation-state: unverified
```

```
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-21
```

```
1:21.21.21.21:1::2222222222222222::0/304
```

```
    *[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
```

```
    AS path: I, validation-state: unverified
```

```
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-21
```

```
1:22.22.22.22:0::2222222222222222::FFFF:FFFF/304
```

```
    *[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
```

```
    AS path: I, validation-state: unverified
```

```
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22
```

```
1:22.22.22.22:1::2222222222222222::0/304
```

```
*[BGP/170] 21:15:38, localpref 100, from 1.1.1.1
  AS path: I, validation-state: unverified
  > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22
```

<snip>

Taking a closer look at the EVPN-1 EVI on PE11 we can see all of the remote PEs, their ESIs, and their Mode discovered via the received Auto-Discovery per ESI advertisements. Note that the output below does not display the Split Horizon label received, only the value that PE11 advertises. Similarly, the Aliasing label from each remote PE's EVPN-1 instance is learned via the received Auto-Discovery per EVI advertisements:

```
cse@PE11> show evpn instance EVPN-1 extensive
Instance: EVPN-1
<snip>
```

```
Number of ethernet segments: 2
ESI: 00:11:11:11:11:11:11:11:11
Status: Resolved by IFL ae0.100
Local interface: ae0.100, Status: Up/Forwarding
Number of remote PEs connected: 1
  Remote PE      MAC label  Aliasing label  Mode
  12.12.12.12    300688     300688          all-active
Designated forwarder: 11.11.11.11
Backup forwarder: 12.12.12.12
Advertised MAC label: 300976
Advertised aliasing label: 300976
Advertised split horizon label: 299984
ESI: 00:22:22:22:22:22:22:22:22
Status: Resolved by NH 1048609
Number of remote PEs connected: 2
  Remote PE      MAC label  Aliasing label  Mode
  21.21.21.21    300848     300848          all-active
  22.22.22.22    301040     301040          all-active
```

The routes below correspond to the two Auto-Discovery routes transmitted by PE21 to PE11. The first route is the Auto-Discovery per ESI route and includes the Route Targets corresponding to EVPN-1 and EVPN-2, the ESI 22222222222222222222222222222222 configured on PE21, the multi-homing all-active mode, and the Split Horizon label 300064.

The second route is the Auto-Discovery per EVI route and includes the Route Target corresponding to EVI EVPN-1. It instructs PE11 to use Aliasing label 300848 when load balancing packets to PE21 for MAC address destinations learned from PE22 that are connected to ESI 22222222222222222222222222222222:

```
cse@PE11> show route table EVPN-1.evpn.0 detail
EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
<snip>
1:21.21.21.21:0::22222222222222222222222222222222::FFFF:FFFF/304 (1 entry, 1 announced)
```

```

*BGP      Preference: 170/-101
          Route Distinguisher: 21.21.21.21:0
          Next hop type: Indirect
          Address: 0x95c594c
          Next-hop reference count: 29
          Source: 1.1.1.1
          Protocol next hop: 21.21.21.21
          Indirect next hop: 0x2 no-forward INH Session ID: 0x0
          State: <Secondary Active Int Ext>
          Local AS: 65000 Peer AS: 65000
          Age: 21:16:54 Metric2: 2
          Validation State: unverified
          Task: BGP_65000.1.1.1.1+179
          Announcement bits (1): 0-EVPN-1-evpn
          AS path: I (Originator)
          Cluster list: 1.1.1.1
          Originator ID: 21.21.21.21
          Communities: target:65000:1 target:65000:2 esi-label:all-
active (label 300064)
          Import Accepted
          Localpref: 100
          Router ID: 1.1.1.1
          Primary Routing Table bgp.evpn.0

1:21.21.21.21:1::2222222222222222:::0/304 (1 entry, 1 announced)
*BGP      Preference: 170/-101
          Route Distinguisher: 21.21.21.21:1
          Next hop type: Indirect
          Address: 0x95c594c
          Next-hop reference count: 29
          Source: 1.1.1.1
          Protocol next hop: 21.21.21.21
          Indirect next hop: 0x2 no-forward INH Session ID: 0x0
          State: <Secondary Active Int Ext>
          Local AS: 65000 Peer AS: 65000
          Age: 21:16:54 Metric2: 2
          Validation State: unverified
          Task: BGP_65000.1.1.1.1+179
          Announcement bits (1): 0-EVPN-1-evpn
          AS path: I (Originator)
          Cluster list: 1.1.1.1
          Originator ID: 21.21.21.21
          Communities: target:65000:1
          Import Accepted
          Route Label: 300848
          Localpref: 100
          Router ID: 1.1.1.1
          Primary Routing Table bgp.evpn.0

```

For further verification of aliasing see the section *Layer 2 Operations - Layer 2 Forwarding with Aliasing* below.

## Inclusive Multicast

Each EVPN PE advertises an Inclusive Multicast (IM) route to enable forwarding of BUM traffic. The IM route advertisement includes:

- The Route Target corresponding to the EVI
- The Ethernet Tag ID – in this case the VLAN ID
- PMSI Tunnel Attribute - indicates the multicast technology to use and related information including the Inclusive Multicast MPLS label

The PMSI Tunnel Attribute is the same attribute that is used in Next Generation BGP Multicast VPNs (<https://tools.ietf.org/html/rfc6513>). It includes the *Tunnel Type* that indicates the multicast technology to be used in the core network to forward BUM traffic. Some examples include ingress replication, P2MP RSVP-TE LSPs, P2MP mLDP LSPs, and PIM-SSM trees.

In the case of ingress replication, when a PE receives a BUM packet from a CE device, it makes a copy of the packet corresponding to each of the remote PEs. It then encapsulates each packet with the appropriate MPLS labels before forwarding the packets. In most cases the ingress PE first pushes the learned Inclusive Multicast label and then pushes a transport label to reach the loopback of the destination PE. The exception to this is a multi-homed non-DF PE sending a BUM packet to its peer DF PE, in which case a Split Horizon label is first pushed (see the *Multi-homing – Auto-Discovery - Auto-Discovery per ESI Route* section above).

After the transport label has been removed from the packet, the receiving PE recognizes the IM label and classifies the packet as BUM traffic. The PE then forwards it appropriately depending on whether or not it is a DF, and based on a check of the Split Horizon label if it is present.

As is often the case in the world of technology, there are trade-offs between the different multicast techniques. The use of P2MP LSPs results in better core bandwidth utilization as the ingress PE transmits a single copy of BUM traffic to the replication point. However, this approach requires maintenance of additional state on the core router that serves the role of replication point for that P2MP LSP, and might not be very scalable considering there would be at least one unique P2MP LSP per EVI.

In order to simplify forwarding in the core while independently scaling the number of EVIs at the edge, the initial implementation of EVPN in Junos supports ingress replication. The trade-off in this case is the



## Lab Example – IM

MAC address advertisement:	3
MAC+IP address advertisement:	1

```

    Inclusive multicast:          1
    Ethernet auto-discovery:      2
21.21.21.21
  Received routes
    MAC address advertisement:    1
    MAC+IP address advertisement: 0
    Inclusive multicast:          1
    Ethernet auto-discovery:      2
22.22.22.22
  Received routes
    MAC address advertisement:    2
    MAC+IP address advertisement: 2
    Inclusive multicast:          1
    Ethernet auto-discovery:      2

```

<snip>

Zooming in on the IM route from PE21 for EVPN-1, we can see the PMSI Tunnel attribute. In the lab topology the data center PEs use Tunnel Type INGRESS-REPLICATION, which means that the PE that receives the BUM packet makes and sends a copy for each of the remote PEs.

The PMSI Tunnel attribute also includes the Tunnel Identifier, which is the loopback IP address of the advertising PE, and the Inclusive Multicast MPLS Label. In this example, when PE11 forwards BUM traffic to PE21 it first pushes the IM label with value 311168 and then pushes the transport label to reach the loopback IP address of PE21:

```

cse@PE11> show route table EVPN-1.evpn.0 detail
<snip>
3:21.21.21.21:1::100::21.21.21.21/304 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Route Distinguisher: 21.21.21.21:1
            PMSI: Flags 0x0: Label 311168: Type INGRESS-REPLICATION 21.21.21.21
            Next hop type: Indirect
            Address: 0xc84252c
            Next-hop reference count: 27
            Source: 1.1.1.1
            Protocol next hop: 21.21.21.21
            Indirect next hop: 0x2 no-forward INH Session ID: 0x0
            State: <Secondary Active Int Ext>
            Local AS: 65000 Peer AS: 65000
            Age: 1d 17:44:24      Metric2: 2
            Validation State: unverified
            Task: BGP_65000.1.1.1.1+179
            Announcement bits (1): 0-EVPN-1-evpn
            AS path: I (Originator)
            Cluster list: 1.1.1.1
            Originator ID: 21.21.21.21
            Communities: target:65000:1
            Import Accepted
            Localpref: 100
            Router ID: 1.1.1.1
            Primary Routing Table bgp.evpn.0

```

<snip>

## Layer 2 Operations

### MAC Learning

When a PE router detects a new MAC address on its EVI access interface, it adds the address to its appropriate local Layer 2 forwarding table, or MAC-VRF. The PE then transmits a MAC Advertisement route using MP-BGP to all remote PEs. This is essentially the control plane-based MAC learning process that is fundamental to EVPN.

The PE's MAC Advertisement route includes the following:

- A Route Target corresponding to the EVI.
- The MAC address that was learned.
- The Ethernet Tag, or VLAN ID, in which the MAC address was learned.
- The ESI on which the MAC address was learned.
- The IP address corresponding to the MAC address, if known and if an IRB is configured.
- An MPLS Service label, or MAC label, corresponding to the MAC address.
- Default Gateway Extended Community – for the MAC/IP address binding that is configured on the VLAN's IRB interface.
- MAC Mobility Extended Community – for processing MAC moves and to detect MAC flapping.

When a PE initially learns a MAC address from its local connection to the CE, it transmits a MAC Advertisement route without the IP address. Once the MAC/IP binding of a given host is learned by the PE, it then transmits another MAC Advertisement route which contains both the MAC and IP addresses. This process is also known as *Host MAC/IP Synchronization*. In this lab network the MAC/IP bindings are dynamically learned by the PE via ARP snooping. Also note that an IRB interface for the EVPN VLAN must be configured in order for the PE to transmit MAC/IP Advertisement routes.

A PE also advertises a MAC/IP Advertisement route containing the IP and MAC address of the locally configured IRB interface along with the Default Gateway Extended Community. The Default Gateway Extended Community signals to the receiving PE that it must route traffic on behalf of the advertising PE. This process is also referred to as *Default Gateway Synchronization*. The MAC/IP Advertisements are essential to the integration of Layer 3 routing with Layer 2 EVPNs,

as we'll explore in more detail in the forthcoming *Layer 3 Operations* section.

The inclusion of the ESI in the MAC Advertisement route is critical for implementing aliasing, or load balancing. In the previous section we learned that multi-homed PEs advertise their connectivity to a common ESI by transmitting Auto-Discovery routes to all remote PEs. When a given remote PE subsequently learns of a MAC address from that ESI, it knows that the destination is reachable via the set of multi-homed PEs. The PE can then load balance traffic to the multiple PEs connected to the common ES. The *Layer 2 Forwarding with Aliasing* section below covers this in more detail.

The ESI in the MAC Advertisement route also ensures that forwarding to local destinations on multi-homed PEs is optimized. When a PE receives a MAC Advertisement route from its multi-homed peer PE, it installs a forwarding entry in the appropriate EVI VLAN's MAC-VRF table with its local ES interface as the next hop. The result is that the PE's local interface is always preferred over the connectivity via the core. This assumes that the MAC address has not already been learned by the PE via its local interface, in which case the PE ignores the received MAC Advertisement route.

When a local MAC address ages out of the PE's forwarding table, it must withdraw the previously advertised MAC Advertisement route since the destination is no longer reachable. The advertisement and withdrawal of MAC routes is especially important in cases where a MAC address moves from one ES to another. The MAC Mobility Extended Community helps ensure that this process is robust. This topic is discussed in more detail in the *MAC Mobility* section below.

### Lab Example – MAC Learning

Viewing the EVI EVPN-1 status on PE11 shows many details related to MAC learning. First, the service label, or MAC route label, advertised to remote PEs is 300944. This instance has received six MAC Advertisement routes from Remote PEs, which includes PE12 and the PEs in Data Center 2, and has learned two local MAC addresses on a Local interface. The instance has one Local default gateway MAC address that corresponds to the statically configured IRB interface. Note that an EVI containing multiple VLANs, for example EVPN-2, displays the aggregate of MAC addresses across all VLANs when displaying the Total MAC addresses.

In addition, PE11 has received MAC address and MAC+IP address advertisements from its neighbors. Under the MAC label heading you

can see that PE11 has also learned the MPLS MAC service label value from each of the remote PEs. Interestingly, each PE advertises a service label and aliasing label with the same label value:

```
cse@PE11> show evpn instance EVPN-1 extensive
Instance: EVPN-1
Route Distinguisher: 11.11.11.11:1
VLAN ID: 100
Per-instance MAC route label: 300944
MAC database status
Total MAC addresses: 2 6
Default gateway MAC addresses: 1 0
Number of local interfaces: 1 (1 up)
Interface name ESI Mode Status
ae0.100 00:11:11:11:11:11:11:11:11:11 all-active Up
Number of IRB interfaces: 1 (1 up)
Interface name VLAN ID Status L3 context
irb.100 100 Up IPVPN-1
Number of bridge domains: 1
VLAN ID Intfs / up Mode MAC sync IM route label
100 1 1 Extended Enabled 301216
Number of neighbors: 3
12.12.12.12
Received routes
MAC address advertisement: 3
MAC+IP address advertisement: 1
Inclusive multicast: 1
Ethernet auto-discovery: 2
21.21.21.21
Received routes
MAC address advertisement: 1
MAC+IP address advertisement: 0
Inclusive multicast: 1
Ethernet auto-discovery: 2
22.22.22.22
Received routes
MAC address advertisement: 2
MAC+IP address advertisement: 2
Inclusive multicast: 1
Ethernet auto-discovery: 2
Number of ethernet segments: 2
ESI: 00:11:11:11:11:11:11:11:11:11
Status: Resolved by IFL ae0.100
Local interface: ae0.100, Status: Up/Forwarding
Number of remote PEs connected: 1
Remote PE MAC label Aliasing label Mode
12.12.12.12 300688 300688 all-active
Designated forwarder: 11.11.11.11
Backup forwarder: 12.12.12.12
Advertised MAC label: 300976
Advertised aliasing label: 300976
Advertised split horizon label: 299984
ESI: 00:22:22:22:22:22:22:22:22:22
Status: Resolved by NH 1048609
```

Number of remote PEs connected: 2

Remote PE	MAC label	Aliasing label	Mode
21.21.21.21	300848	300848	all-active
22.22.22.22	301040	301040	all-active

The Layer 2 forwarding table for the EVI EVPN-1 shows the dynamically learned MAC addresses:

```
cse@PE11> show evpn mac-table
```

MAC flags (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC, O -OVSDB MAC, SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

Routing instance : EVPN-1

Bridging domain : \_\_EVPN-1\_\_, VLAN : 100

MAC address	MAC flags	Logical interface	NH Index	RTR ID
00:00:09:c1:b0:d3	D	ae0.100		
00:00:09:c1:b0:d7	DC		1048609	1048609
00:00:51:01:96:44	D	ae0.100		
00:00:51:01:96:45	DRC	ae0.100		
00:00:51:01:96:46	DRC	ae0.100		
00:50:56:8c:76:67	DC		1048609	1048609

You can see that two of the MAC addresses, with MAC flags values of only D, are locally learned on access interface ae0.100. The other four are learned from Remote PEs, including PE12 and the PEs at Data Center 2. These entries all have a MAC flags value of C indicating they are Control MACs learned via MP-BGP MAC Advertisement route updates. In addition, two of the four entries have a MAC flags value of R indicating that they are learned from the multi-homed peer PE12, and are reachable via the local access interface ae0.100.

Why are there only four remotely learned MAC addresses when the Total MAC Addresses output above indicated that there are six? The reason for this is that two of the MAC Advertisement routes received by PE11, from PE12, match the two locally learned MAC addresses. The two local MAC addresses are preferred, therefore, the corresponding MAC Advertisement routes are ignored.

**NOTE** The equivalent command for an EVI configured as a Virtual Switch is `show bridge mac-table bridge-domain <VLAN name> instance <EVI name>`.

Recall that the IP address corresponding to a host's MAC address is also advertised when known. The evpn database for EVI EVPN-1 includes the same MAC addresses as the Layer 2 forwarding table, their associated IP addresses if known, and ESI location. An entry corresponding to the locally configured default gateway is also listed:

```
cse@PE11> show evpn database instance EVPN-1
Instance: EVPN-1
```

VLAN	MAC address	Active source	Timestamp	IP address
100	00:00:00:01:01:01	irb.100	Nov 06 17:54:46	100.1.1.1
100	00:00:09:c1:b0:d3	00:11:11:11:11:11:11:11:11:11	Nov 07 09:45:31	100.1.1.12
100	00:00:09:c1:b0:d7	00:22:22:22:22:22:22:22:22:22	Nov 07 09:45:32	100.1.1.29
100	00:00:51:01:96:44	00:11:11:11:11:11:11:11:11:11	Nov 07 09:54:53	
100	00:00:51:01:96:45	00:11:11:11:11:11:11:11:11:11	Nov 07 09:54:55	
100	00:00:51:01:96:46	00:11:11:11:11:11:11:11:11:11	Nov 07 09:54:55	
100	00:50:56:8c:76:67	00:22:22:22:22:22:22:22:22:22	Nov 07 07:12:51	100.1.1.10

Zooming in on MAC address 00:50:56:8c:76:67 shows that there are three MAC Advertisement routes learned from remote PEs. This MAC address corresponds to the VLAN 100 virtual machine (VM) that is currently in Data Center 2. Both PE21 and PE22 have learned the MAC address of the host and have advertised it to PE11. In addition, PE22 advertises the MAC/IP address binding. Note that these routes are EVPN NLRI Route Type 2:

```
cse@PE11> show route table EVPN-1.evpn.0 evpn-mac-address 00:50:56:8c:76:67
```

```
EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
2:21.21.21.21:1::100::00:50:56:8c:76:67/304
    *[BGP/170] 04:03:50, localpref 100, from 1.1.1.1
    AS path: I, validation-state: unverified
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-21
2:22.22.22.22:1::100::00:50:56:8c:76:67/304
    *[BGP/170] 1d 01:03:50, localpref 100, from 1.1.1.1
    AS path: I, validation-state: unverified
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22
2:22.22.22.22:1::100::00:50:56:8c:76:67::100.1.1.10/304
    *[BGP/170] 1d 01:03:50, localpref 100, from 1.1.1.1
    AS path: I, validation-state: unverified
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22
```

An even closer look at the last MAC/IP Advertisement route shows that the Route Target matches EVPN-1, the VLAN ID is 100, the service label for PE11 to use to send traffic to PE22 for this MAC destination is 301040, and the MAC address was learned via ESI 00:22:22:22:22:22:22:22:22:22 which is configured in Data Center 2:

```
cse@PE11> show route table EVPN-1.evpn.0 evpn-mac-address 00:50:56:8c:76:67 detail
```

```
EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
<snip>
2:22.22.22.22:1::100::00:50:56:8c:76:67::100.1.1.10/304 (1 entry, 1 announced)
    *BGP Preference: 170/-101
    Route Distinguisher: 22.22.22.22:1
    Next hop type: Indirect
    Address: 0x95c5868
    Next-hop reference count: 41
```

```

Source: 1.1.1.1
Protocol next hop: 22.22.22.22
Indirect next hop: 0x2 no-forward INH Session ID: 0x0
State: <Secondary Active Int Ext>
Local AS: 65000 Peer AS: 65000
Age: 1d 16:51:15 Metric2: 2
Validation State: unverified
Task: BGP_65000.1.1.1.1+179
Announcement bits (1): 0-EVPN-1-evpn
AS path: I (Originator)
Cluster list: 1.1.1.1
Originator ID: 22.22.22.22
Communities: target:65000:1
Import Accepted
Route Label: 301040
ESI: 00:22:22:22:22:22:22:22:22:22
Localpref: 100
Router ID: 1.1.1.1
Primary Routing Table bgp.evpn.0

```

## Layer 2 Forwarding with Aliasing

As mentioned in the preceding *MAC Learning* section, aliasing is made possible by the inclusion of the ESI value in the Auto-Discovery and MAC routes advertised by multi-homed PEs. This information is used by a given remote PE to load balance traffic to the destination MAC address by sending traffic to the set of multi-homed PEs connected to the ES. First, let's take a closer look at the forwarding tables to get a better understanding of how this is implemented in Junos. Then we'll send some traffic flows to verify that traffic is load balanced.

### Lab Example - Aliasing

Now that the VLAN forwarding tables have been populated with MAC addresses, a closer examination of the forwarding operation can be performed. The mac-table for EVPN-1 on PE11 shows that the next hop index value for MAC address destinations in Data Center 2 is 1048609:

```
cse@PE11> show evpn mac-table
```

```
MAC flags      (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
O -OVSDB MAC, SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)
```

```
Routing instance : EVPN-1
```

```
Bridging domain : __EVPN-1__, VLAN : 100
```

MAC address	MAC flags	Logical interface	NH Index	RTR ID
00:00:09:c1:b0:d3	D	ae0.100		
00:00:09:c1:b0:d7	DC		1048609	1048609
00:00:51:01:96:44	D	ae0.100		
00:00:51:01:96:45	DRC	ae0.100		
00:00:51:01:96:46	DRC	ae0.100		
00:50:56:8c:76:67	DC		1048609	1048609



This value represents an index to a list of next hops corresponding to the destination PEs. To determine the list of destination PEs, first find the ESI associated with the next hop index. In this case, the next hop index matches the ESI configured at Data Center 2:

```
cse@PE11> show evpn instance EVPN-1 extensive
Instance: EVPN-1
  Route Distinguisher: 11.11.11.11:1
  VLAN ID: 100
<snip>
  Number of ethernet segments: 2
<snip>
  ESI: 00:22:22:22:22:22:22:22:22:22
  Status: Resolved by NH 1048609
  Number of remote PEs connected: 2


| Remote PE   | MAC label | Aliasing label | Mode       |
|-------------|-----------|----------------|------------|
| 21.21.21.21 | 300848    | 300848         | all-active |
| 22.22.22.22 | 301040    | 301040         | all-active |


```

Next, find the MPLS label associated with the ESI in the mpls.0 table. In this case the MPLS label is 301200. Note that the MPLS label in this case is not used for traffic forwarding. It represents a dummy route that is used by EVPN to point to a list of next hops:

```
cse@PE11> show route table mpls.0 | match EVPN-1 | match esi
301184          *[EVPN/7] 2d 03:35:55, routing-instance EVPN-1, route-type Egress-
MAC, ESI 00:11:11:11:11:11:11:11:11:11
301200          *[EVPN/7] 2d 03:35:55, routing-instance EVPN-1, route-type Egress-
MAC, ESI 00:22:22:22:22:22:22:22:22:22
```

Take a look at the mpls.0 table again to view the next hops for the label entry. In this case they are PE21 and PE22. Looking at the forwarding-table shows that both next hops are actively used since EVPN load balancing, or aliasing, is enabled by default. The labels 300848 and 301040 correspond to the MAC and Aliasing labels received from the remote PEs:

```
cse@PE11> show route label 301200

mpls.0: 37 destinations, 38 routes (37 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

301200          *[EVPN/7] 2d 03:44:05, routing-instance EVPN-1, route-type Egress-
MAC, ESI 00:22:22:22:22:22:22:22:22:22
    > to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-21
    to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22

cse@PE11> show route forwarding-table label 301200
Routing table: default.mpls
MPLS:


| Destination | Type | RtRef | Next hop | Type | Index   | NhRef | Netif |
|-------------|------|-------|----------|------|---------|-------|-------|
| 301200      | user | 0     |          | ulst | 1048609 | 2     |       |
|             |      |       |          | indr | 1048577 | 3     |       |


```

```

10.11.1.1      Push 300848, Push 300912(top) 606 3 xe-
1/2/0.0
                                indr 1048574 3
10.11.1.1      Push 301040, Push 300928(top) 603 3 xe-
1/2/0.0

```

To verify the EVPN aliasing feature, four Layer 2 traffic flows are transmitted at 1000 packets per second (pps) from the each of the four emulated hosts on Ixia 9/12 to the destination host Ixia 9/29, which is configured with IP address 100.1.1.29 and MAC address 00:00:09:c1:b0:d7. Based on the output of the EVPN-1 forwarding mac-table above, this destination has the next hop index of 1048609, which we've just confirmed load balances traffic to PE21 and PE22.

Once the flows are started CE10 shows that the four traffic flows are equally load balanced amongst the two uplinks:

```

CE10                               Seconds: 40                               Time: 04:25:14
Interface  Link  Input packets  (pps)  Output packets  (pps)
gr-0/0/0   Up    0              (0)    0              (0)
pfh-0/0/0   Up    0              (0)    0              (0)
xe-0/0/0    Up    31872          (1)    4784900        (1998)
xe-0/0/1    Up    110810         (0)    4757462        (2000)
xe-0/0/2    Up    193306         (0)    200358         (0)
xe-0/0/9    Up    1736           (0)    67306          (0)
xe-0/0/10   Up    1740           (0)    67448          (0)
xe-0/0/11   Up    1740           (0)    67292          (0)
xe-0/0/12   Up    9351639        (3999)  69788          (1)
xe-0/0/30   Up    4              (0)    51360          (0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

```

The LSP statistics on PE11 show that the traffic flows it receives are load balanced to PE21 and PE22:

```
cse@PE11> clear mpls lsp statistics
```

```
cse@PE11> show mpls lsp statistics ingress
```

```
Ingress LSP: 4 sessions
```

To	From	State	Packets	Bytes	LSPname
12.12.12.12	11.11.11.11	Up	1	74	from-11-to-12
21.21.21.21	11.11.11.11	Up	5966	1527296	from-11-to-21
22.22.22.22	11.11.11.11	Up	5965	1527040	from-11-to-22
31.31.31.31	11.11.11.11	Up	0	0	from-11-to-31

Total 4 displayed, Up 4, Down 0

Similar forwarding behavior is observed on PE12:

```
cse@PE12> clear mpls lsp statistics
```

```
cse@PE12> show mpls lsp statistics ingress
```

```
Ingress LSP: 4 sessions
```

To	From	State	Packets	Bytes	LSPName
11.11.11.11	12.12.12.12	Up	0	0	from-12-to-11
21.21.21.21	12.12.12.12	Up	3977	1018112	from-12-to-21
22.22.22.22	12.12.12.12	Up	3977	1018112	from-12-to-22
31.31.31.31	12.12.12.12	Up	0	0	from-12-to-31

Total 4 displayed, Up 4, Down 0

The statistics on CE20 show that the traffic flows, 4000 pps total, are received from PE21 and PE22 and delivered to the Ixia 9/29 host on interface xe-0/0/23:

```
CE20                               Seconds: 83                               Time: 00:45:35
```

Interface	Link	Input packets	(pps)	Output packets	(pps)
gr-0/0/0	Up	0	(0)	0	(0)
pfh-0/0/0	Up	0		0	
xe-0/0/0	Up	4227669242	(2000)	2692511762	(0)
xe-0/0/1	Up	3627401176	(2000)	4483966194	(0)
xe-0/0/2	Up	1531670098	(0)	1530088394	(0)
xe-0/0/20	Up	1411490783	(0)	1840893097	(0)
xe-0/0/21	Up	1410359171	(0)	1409884454	(0)
xe-0/0/22	Up	1412632510	(0)	1412218002	(0)
xe-0/0/23	Up	1410407633	(1)	1662016101	(4000)
xe-0/0/30	Up	196655	(0)	217656	(0)
ae0	Up	7855070418	(4000)	7176477956	(0)
bme0	Up	0		6774	

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

In this example there are multiple levels of load balancing taking place. First, CE10 load balances the traffic flows across the all-active multi-homed uplinks to PE11 and PE12. These PE1s then load balance traffic to the remote PE1s in Data Center 2 using EVPN aliasing.

In practice there are additional levels of load balancing that can take place. For example, if there are multiple LSPs between a pair of PE1s then traffic flows to the destination PE1 would be load balanced. If an LSP happens to traverse a LAG then traffic flows would be further load balanced amongst the link bundle members. Thus, with EVPN optimal link utilization is achieved in all segments of the network including access and core for Layer 2 as well as Layer 3 traffic, which we'll see in the *Layer 3 Operations - Inter-VLAN Routing* section.

Note that load balancing in the EVPN is performed at the flow level. This means that all packets of a given flow take the same path through the network. This eliminates the possibility of packet reordering and minimizes jitter. A flow is identified based on the fields present in the packet header and Junos provides fine control over which fields to use for identification.

## MAC Mobility

As discussed previously, when a PE learns of a source MAC address via its local EVI access interface, it transmits a MAC Advertisement route to all other PEs. However, if the MAC address moves to another Ethernet Segment, such that it is now reachable via another PE, the original PE may not be aware of this action and will not withdraw the MAC Advertisement route. EVPN is designed to address this scenario by using the MP-BGP control plane to track the movement of MAC addresses, also known as *MAC Mobility*.

Suppose a MAC address moves to another Ethernet Segment in a remote data center. When the PE in the new data center learns the MAC address it transmits a MAC Advertisement route to all other PEs. The PE router in the original data center receives this route and takes a few actions. First, it updates its forwarding table with the new reachability information, which then triggers the withdrawal of its previously advertised MAC Advertisement route.

As mentioned in the preceding *MAC Learning* section, the MAC Mobility Extended Community includes a sequence number that increments with each MAC move. This is used to by PEs to ensure that the MAC Advertisements are processed correctly. It can also be used to detect MAC flapping. For example, if a PE detects that a number of MAC moves within a given time period exceeds a configured threshold it can alert the network operator and stop sending MAC Advertisement routes. This community will be supported in an upcoming release of Junos.

### Lab Example – MAC Mobility

In this book's lab topology there is a VM in VLAN 100 running on Server 2 in Data Center 2 with the MAC address 00:50:56:8C:76:67. The EVPN VLAN 100 forwarding table on PE11 shows that this MAC address is remote. As seen previously, PE11 receives MAC Advertisement routes for this MAC address from PE21 and PE22. Note that the output on PE12 is similar:

```
cse@PE11> show evpn mac-table 00:50:56:8c:76:67
```

```
MAC flags          (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
O -OVSDB MAC, SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)
```

```
Routing instance : EVPN-1
```

```
Bridging domain : __EVPN-1__, VLAN : 100
```

MAC address	MAC flags	Logical interface	NH Index	RTR ID
00:50:56:8c:76:67	DC		1048609	1048609

On PE21 this MAC address is learned locally. Note that the output on PE22 is similar:

```
cse@PE21> show evpn mac-table 00:50:56:8c:76:67
```

```
MAC flags      (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
O -OVSDB MAC, SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)
```

```
Routing instance : EVPN-1
```

```
Bridging domain : __EVPN-1__, VLAN : 100
```

MAC address	MAC flags	Logical interface	NH Index	RTR ID
<b>00:50:56:8c:76:67</b>	<b>D</b>	<b>ae0.100</b>		

Next, let's move the VM to Server 1 in Data Center 1 using VMware vMotion. The MAC address 00:50:56:8c:76:67 is now local to PE11 interface ae0.100:

```
cse@PE11> show evpn mac-table 00:50:56:8c:76:67
```

```
MAC flags      (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
O -OVSDB MAC, SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)
```

```
Routing instance : EVPN-1
```

```
Bridging domain : __EVPN-1__, VLAN : 100
```

MAC address	MAC flags	Logical interface	NH Index	RTR ID
<b>00:50:56:8c:76:67</b>	<b>D</b>	<b>ae0.100</b>		

On PE11 we can see that the MAC Advertisement routes from PE21 and PE22 have been withdrawn. Only a single MAC route from PE12 is received and ignored:

```
cse@PE11> show route table EVPN-1.evpn.0 evpn-mac-address 00:50:56:8c:76:67
```

```
EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
2:11.11.11.11:1::100::00:50:56:8c:76:67/304
```

```
*[EVPN/170] 00:02:13
```

```
Indirect
```

```
2:12.12.12.12:1::100::00:50:56:8c:76:67/304
```

```
*[BGP/170] 00:02:11, localpref 100, from 1.1.1.1
```

```
AS path: I, validation-state: unverified
```

```
> to 10.11.12.12 via ae1.0, label-switched-path from-11-to-12
```

```
2:11.11.11.11:1::100::00:50:56:8c:76:67::100.1.1.10/304
```

```
*[EVPN/170] 00:02:12
```

```
Indirect
```

## Layer 3 Operations

Although EVPN is a Layer 2 VPN technology, the hosts on an EVPN VLAN ultimately need to communicate with devices in other local or remote subnets. In the sections below we'll use our lab topology to take a closer look at Default Gateway Synchronization, which enables optimization of outbound routing. Then we'll focus on two other scenarios in which traffic is efficiently routed due to Host MAC/IP Synchronization, which was discussed briefly in the preceding *MAC Learning* section. First, we'll see how traffic between EVPN VLANs is efficiently routed between PEs. Then we'll move a VM between data centers and verify that inbound traffic from a remote IP VPN site to the VM is always optimally routed.

### Default Gateway Synchronization

Outbound routing refers to traffic flows originating from hosts in the EVPN VLANs, or in the data centers. In an environment where VMs can be easily migrated between data centers, a feature called *Default Gateway Synchronization* allows the PE that is local to the VM to always route outbound traffic. This provides optimal routing from the data center to any remote destination and eliminates the need to backhaul traffic to a migrated VM's original default gateway.

Optimized outbound routing is accomplished by first configuring an IRB interface on the VLAN, or bridge domain, associated with the EVPN. This IRB interface is then associated with an IP VPN. Since the IRB interface is used as the default gateway, hosts can route to any destinations reachable via the IP VPN's forwarding table, or VRF. As we'll see in the *Inter-VLAN Routing* section below, this integration between EVPN and IP VPN also enables the use of aliasing when an EVPN PE routes traffic to a destination host reachable via remote multi-homed EVPN PEs.

#### Lab Example - Default Gateway Synchronization

In the lab topology each of the four VLANs (100, 200-202) are configured with an IRB interface (as discussed previously in *Chapter 2 Configuration - Services*). Each of these IRB interfaces is placed into a common IP VPN, named *IPVPN-1*. PE31 is also a member of *IPVPN-1* and provides connectivity to a remote site. Therefore, hosts on the EVPN VLANs can communicate with each other and to destinations at the remote site.

Once the IRB interface is configured its MAC/IP binding is transmitted to all other PEs via an EVPN MAC/IP Advertisement route that

contains the Default Gateway Extended Community. This specific advertisement is how the default gateway information between PEs gets “synchronized.”

Recall from the EVI EVPN-1 configuration that the same MAC/IP address is configured for the VLAN 100 IRB on all PEs. In this case there is no need to advertise the MAC/IP binding as the default gateway information is essentially statically synchronized. By setting the `evpn default-gateway do-not-advertise` parameter under the `routing-instances` configuration the advertisement is suppressed. For example, on PE11 there are no MAC/IP Advertisement routes received with a MAC/IP binding corresponding to the default gateway address of 100.1.1.1:

```
cse@PE11> show route table EVPN-1.evpn.0 | match "100.1.1.1/"
```

However, PE11 does receive MAC/IP Advertisement routes corresponding to the default gateways of the three VLANs in EVPN-2:

```
cse@PE11> show route table EVPN-2.evpn.0 | match "200.1.1.1/"
2:22.22.22.22:2::200::00:00:c8:01:01:01::200.1.1.1/304
```

```
cse@PE11> show route table EVPN-2.evpn.0 | match "201.1.1.2/"
2:22.22.22.22:2::201::00:00:c9:01:01:02::201.1.1.2/304
```

```
cse@PE11> show route table EVPN-2.evpn.0 | match "202.1.1.1/"
2:22.22.22.22:2::202::00:00:ca:01:01:01::202.1.1.1/304
```

In EVI EVPN-2, VLAN 201 is configured with different default gateway addresses at each data center. As a result, PE11 accepts the MAC/IP Advertisement routes for VLAN 201 containing the default gateway information of the PE routers at Data Center 2. This enables PE11 to perform Proxy ARP for the default gateway IP address of the remote PE’s IRB interface, responding with the remote PE’s MAC address. PE11 will also route packets destined to the learned default gateway’s MAC address. PE21 behaves similarly based on the MAC/IP Advertisement routes received from PE11 and PE12 for EVPN-2 VLAN 201.

From the CLI we can confirm that PE11 installs the MAC address 00:00:c9:01:01:02 which is configured on the VLAN 201 IRB interface on PE21 and PE22:

```
cse@PE11> show bridge evpn peer-gateway-macs
```

```
Routing instance : EVPN-2
Bridging domain : V201, VLAN : 201
Installed GW MAC addresses:
00:00:c9:01:01:02
```

Similarly, PE21 installs the MAC address 00:00:c9:01:01:01, which is configured on the VLAN 201 IRB interface on PE11 and PE12:

```
cse@PE21> show bridge evpn peer-gateway-macs
```

```
Routing instance : EVPN-2
Bridging domain : V201, VLAN : 201
Installed GW MAC addresses:
00:00:c9:01:01:01
```

Here is a closer look at the MAC/IP Advertisement route for the VLAN 201 default gateway received by PE11 from PE22. It includes the Route Target corresponding to EVI EVPN-2, the VLAN ID 201, the MAC/IP binding, a service label, and the important `evpn-default-gateway` community:

```
cse@PE11> show route table EVPN-2.evpn.0 detail evpn-ethernet-tag-id 201 evpn-mac-address 00:00:c9:01:01:02
```

```
EVPN-2.evpn.0: 40 destinations, 40 routes (40 active, 0 holddown, 0 hidden)
<snip>
2:22.22.22.22:2::201::00:00:c9:01:01:02::201.1.1.2/304 (1 entry, 1 announced)
    *BGP      Preference: 170/-101
              Route Distinguisher: 22.22.22.22:2
              Next hop type: Indirect
              Address: 0x95c5868
              Next-hop reference count: 37
              Source: 1.1.1.1
              Protocol next hop: 22.22.22.22
              Indirect next hop: 0x2 no-forward INH Session ID: 0x0
              State: <Secondary Active Int Ext>
              Local AS: 65000 Peer AS: 65000
              Age: 1d 22:31:01      Metric2: 2
              Validation State: unverified
              Task: BGP_65000.1.1.1.1+179
              Announcement bits (1): 0-EVPN-2-evpn
              AS path: I (Originator)
              Cluster list: 1.1.1.1
              Originator ID: 22.22.22.22
              Communities: target:65000:2 evpn-default-gateway
              Import Accepted
              Route Label: 300288
              ESI: 00:00:00:00:00:00:00:00:00
              Localpref: 100
              Router ID: 1.1.1.1
              Primary Routing Table bgp.evpn.0
```

We will verify the Default Gateway Synchronization feature in the *Lab Example - Inbound and Outbound Routing with MAC Mobility* section below.



## Inter-VLAN Routing

The integration of EVPN with IP VPN is used to efficiently route traffic between hosts or devices on different EVPN VLANs, connected to different PE routers. A feature called *Asymmetric IRB Forwarding* allows an ingress PE, which serves as the source device's default gateway, to forward packets in such a way that eliminates the need to perform a route lookup in the IP VPN VRF of the egress PE. Instead, the egress PE simply performs a lookup in the egress MAC-VRF corresponding to the destination host's EVI. This "asymmetry" between table lookups on ingress and egress PEs is illustrated in Figure 3.2.

Note that the ingress and egress PEs could be in the same or in different data centers. In addition, if the egress PEs are multi-homed then the ingress PE can use aliasing to load balance traffic to the destination.

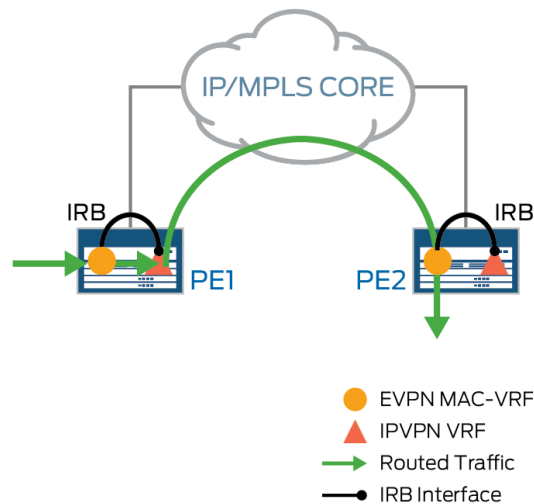


Figure 3.2 Asymmetric IRB Forwarding

### Implementation

As discussed in the *Layer 2 Operations – MAC Learning* section above, EVPN PEs dynamically learn MAC/IP bindings, for example via ARP snooping. The binding information is then transmitted to remote PEs using MAC/IP Advertisement routes, also known as Host MAC/IP Synchronization, and is the key to implementing asymmetric IRB forwarding on the ingress PE.

Let's take a closer look at the actions a PE router takes when it learns a new MAC/IP binding:

- The PE installs a host route in the IP VPN VRF with protocol type EVPN and a next hop of the VLAN's IRB interface. This triggers the PE to transmit this host route to remote PEs that are members

of the IP VPN via a VPN route advertisement. The remote PEs add this route to their VRF with protocol type BGP.

- The PE also advertises the MAC/IP binding via a MAC/IP Advertisement route to remote PEs. A remote data center PE installs a corresponding host route in its IP VPN VRF with protocol type EVPN. It also updates the MAC address information in the MAC-VRF table, if necessary, as well as the Layer 2 information associated with the EVPN host route in the IP VPN VRF. This Layer 2 information is used for asymmetric IRB forwarding.

As a result of these actions a remote data center PE populates its IP VPN VRF with two host routes with different protocol types. The first route is a standard IP VPN route with protocol type BGP. The second route is dynamically generated as a result of the received MAC/IP Advertisement route and has protocol type EVPN.

Both of these routes allow the ingress PE to forward traffic to the destination host, so what is the difference between them? The BGP route provides reachability using the standard IP VPN forwarding mechanism. Specifically, the VPN label is first pushed on the packet and then the transport label, or tunnel label, is pushed to reach the destination PE.

The EVPN route, on the other hand, uses a *composite next hop*, which means that the ingress PE takes multiple actions before forwarding the packet. Using the Layer 2 information contained in the MAC/IP Advertisement associated with the route, it first rewrites the Ethernet header of the packet. This includes setting the destination MAC address and VLAN ID to the values corresponding to the destination host. Also, based on standard routing procedure, the source MAC address of the Ethernet header is set to the local IRB's MAC address.

The ingress PE then pushes the EVPN service or aliasing label followed by the transport label. The EVPN label instructs the egress PE to forward the packet using the EVPN MAC-VRF corresponding to the destination VLAN. This is how the packet bypasses the IP VPN VRF on egress. In addition, because the ingress PE rewrote the Layer 2 header, the packet is then forwarded without any modification to the destination host based on the Destination MAC address.

On the data center PEs the protocol type BGP routes, learned via the IP VPN, are redundant. Therefore, to reduce overhead you can optionally discard these routes by applying policies to the IP VPN instance's VRF. An example of this is configured in the *Chapter 2* section: *Configuration - Services*, where a unique community is added to the advertised IP VPN routes such that they are discarded when received by remote data center PEs.

### Lab Example - Inter-VLAN Routing Between Data Centers

Let's take a look at an example in our lab topology to help make this clear. Host 200.1.1.26 currently resides on VLAN 200 in Data Center 2 and its MAC/IP binding is learned by PE21 EVI EVPN-2. The corresponding EVPN host route is in the IP VPN-1 VRF and the next hop is set to PE21's local `irb.200` interface:

```
cse@PE21> show route 200.1.1.26
```

```
IPVPN-1.inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
200.1.1.26/32      *[EVPN/7] 00:04:52
> via irb.200
```

On PE11 the host route is learned via protocol EVPN from PE21 and installed in the IPVPN-1 VRF. There are two next hops to PE21 and PE22, due to aliasing. Note that PE21 also advertises the host route via the IP VPN, however the configured IP VPN VRF import policy on PE11 discards this redundant route. If this policy were not in place PE11 would have multiple routes for the same destination learned from protocols EVPN and BGP, in which case the EVPN learned route would be preferred due to the lower route preference of "7" for EVPN versus "170" for BGP:

```
cse@PE11> show route 200.1.1.26
```

```
IPVPN-1.inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
200.1.1.26/32      *[EVPN/7] 00:49:50
> to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-21
  to 10.11.1.1 via xe-1/2/0.0, label-switched-path from-11-to-22
```

The route details show the composite next hop whereby PE11 rewrites the Ethernet header of the packet and also pushes the service/aliasing label followed by the transport label to reach the destination PEs in Data Center 2. You can see the Layer 2 Ethernet header rewrite information, which is populated with values from the received MAC/IP Advertisement route. The service/aliasing label instructs PE21 and PE22 to forward the packet using their local EVPN-2 MAC-VRF table:

```
cse@PE11> show route 200.1.1.26 detail
```

```
IPVPN-1.inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
200.1.1.26/32 (1 entry, 1 announced)
```

```
  *EVPN    Preference: 7
           Next hop type: Indirect
           Address: 0x9728448
           Next-hop reference count: 2
```

```

Next hop type: Router, Next hop index: 688
Next hop: 10.11.1.1 via xe-1/2/0.0, selected
Label-switched-path from-11-to-21
Label operation: Push 300256, Push 302448(top)
Label TTL action: no-prop-ttl, no-prop-ttl(top)
Load balance label: Label 300256: None; Label 302448: None;
Session Id: 0x152
Next hop type: Router, Next hop index: 696
Next hop: 10.11.1.1 via xe-1/2/0.0
Label-switched-path from-11-to-22
Label operation: Push 300320, Push 302464(top)
Label TTL action: no-prop-ttl, no-prop-ttl(top)
Load balance label: Label 300320: None; Label 302464: None;
Session Id: 0x152
Protocol next hop: 21.21.21.21
Label operation: Push 300256
Label TTL action: no-prop-ttl
Load balance label: Label 300256: None;
Composite next hop: 0x9569bac 643 INH Session ID: 0x150
Ethernet header rewrite:
  SMAC: 00:00:c8:01:01:01, DMAC: 00:00:09:c1:b0:d4
  TPID: 0x8100, TCI: 0x00c8, VLAN ID: 200, Ethertype: 0x0800
Indirect next hop: 0x96ae310 1048596 INH Session ID: 0x150
Protocol next hop: 22.22.22.22
Label operation: Push 300320
Label TTL action: no-prop-ttl
Load balance label: Label 300320: None;
Composite next hop: 0x9569b50 645 INH Session ID: 0x14f
Ethernet header rewrite:
  SMAC: 00:00:c8:01:01:01, DMAC: 00:00:09:c1:b0:d4
  TPID: 0x8100, TCI: 0x00c8, VLAN ID: 200, Ethertype: 0x0800
Indirect next hop: 0x96ae200 1048597 INH Session ID: 0x14f
State: <Active NoReadvrt Int Ext>
Age: 1:22:10 Metric2: 2
Validation State: unverified
Task: EVPN-2-evpn
Announcement bits (1): 0-KRT
AS path: I

```

On PE11 we can check the `evpn` database for EVI EVPN-2 VLAN 200. The output shows that the MAC address for the 200.1.1.26 host is learned from PE21 and matches the destination MAC address, or `DMAC`, in the route output above. This is important because PE11 rewrites the Layer 2 header of a packet destined to the host using this value. The egress PEs, PE21 and PE22, do not perform any ARP operation, they simply forward the packet based on the destination MAC address:

```
cse@PE11> show evpn database instance EVPN-2 vlan-id 200
```

Instance: EVPN-2

VLAN	MAC address	Active source	Timestamp	IP address
200	00:00:09:c1:b0:d0	00:11:11:11:11:11:11:11:11	Nov 07 18:38:35	200.1.1.9

```

200 00:00:09:c1:b0:d4 00:22:22:22:22:22:22:22:22:22 Nov 07 14:03:35 200.1.1.26
200 00:00:c8:01:01:01 irb.200 Nov 06 17:54:46 200.1.1.1

```

```

cse@PE11> show evpn database instance EVPN-2 vlan-id 200 extensive
Instance: EVPN-2
<snip>
VLAN ID: 200, MAC address: 00:00:09:c1:b0:d4
Source: 00:22:22:22:22:22:22:22:22:22, Rank: 1, Status: Active
Remote origin: 21.21.21.21
Timestamp: Nov 07 14:03:35 (0x5499bc87)
State: <Local-Adv-Allowed Local-Adv-Done>
IP address: 200.1.1.26
Remote origin: 21.21.21.21
L3 route: 200.1.1.26/32, L3 context: IPVPN-1 (irb.200)
<snip>

```

On PE11 we can also verify that the service, or MAC, and aliasing labels associated with the host route match the labels received from the remote PEs in Data Center 2. In this case the MAC label and Aliasing label values in the EVPN instance output below match the first labels that are pushed onto packets destined to 200.1.1.26:

```

cse@PE11> show evpn instance EVPN-2 extensive | find segments
Number of ethernet segments: 2
<snip>
ESI: 00:22:22:22:22:22:22:22:22:22
Status: Resolved by NH 1048590
Number of remote PEs connected: 2
  Remote PE      MAC label  Aliasing label  Mode
  22.22.22.22    300320     300320          all-active
  21.21.21.21    300256     300256          all-active
<snip>

```

To confirm that aliasing will take place we can check the PFE forwarding-table. The PE11 output below shows that both Nexthops are programmed in the PFE because of the default EVPN per-packet load balancing policy applied by Junos:

```

cse@PE11> show route forwarding-table destination 200.1.1.26 extensive | find IPVPN-1
Routing table: IPVPN-1.inet [Index 7]
Internet:

Destination: 200.1.1.26/32
Route type: user
Route reference: 0
Multicast RPF nh index: 0
Flags: sent to PFE
Next-hop type: unicast
Next-hop:
  Next-hop type: composite
  Next-hop type: indirect
Route interface-index: 0
Index: 1048601 Reference: 1
Index: 643 Reference: 2
Index: 1048596 Reference: 4
Weight: 0x0

```

```

Nexthop: 10.11.1.1
Next-hop type: Push 300256, Push 302448(top) Index: 688 Reference: 3
Load Balance Label: None
Next-hop interface: xe-1/2/0.0    Weight: 0x0
Nexthop:
Next-hop type: composite          Index: 645      Reference: 2
Next-hop type: indirect          Index: 1048597 Reference: 4
                                Weight: 0x0

Nexthop: 10.11.1.1
Next-hop type: Push 300320, Push 302464(top) Index: 696 Reference: 3
Load Balance Label: None
Next-hop interface: xe-1/2/0.0    Weight: 0x0

```

Now let's send some traffic flows between data centers to verify aliasing for inter-VLAN routing. From the Ixia port 9/12 in Data Center 1 start four flows, at 1000 pps each, from the four VLAN 100 hosts. The destination of the flows is VLAN 200 host 200.1.1.26 at Ixia port 9/26, which is located in Data Center 2.

Based on the output traffic rate on CE10's two uplinks, the flows are equally distributed to PE11 and PE12:

CE10		Seconds: 28		Time: 04:33:42	
Interface	Link	Input packets	(pps)	Output packets	(pps)
gr-0/0/0	Up	0	(0)	0	(0)
pfh-0/0/0	Up	0		0	
xe-0/0/0	Up	32007	(0)	5237589	(2001)
xe-0/0/1	Up	110964	(0)	5210171	(2000)
xe-0/0/2	Up	193446	(0)	200509	(0)
xe-0/0/9	Up	1738	(0)	67328	(0)
xe-0/0/10	Up	1742	(0)	67470	(0)
xe-0/0/11	Up	1742	(0)	67315	(0)
xe-0/0/12	Up	10257804	(4000)	69969	(0)
xe-0/0/30	Up	4	(0)	51360	(0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

A check of the MPLS LSP statistics on PE11 shows that the traffic flows are load balanced to PE21 and PE22:

```
cse@PE11> clear mpls lsp statistics
```

```
cse@PE11> show mpls lsp statistics ingress
```

```
Ingress LSP: 4 sessions
```

To	From	State	Packets	Bytes	LSPname
12.12.12.12	11.11.11.11	Up	2	148	from-11-to-12
21.21.21.21	11.11.11.11	Up	9942	2545152	from-11-to-21
22.22.22.22	11.11.11.11	Up	9943	2545408	from-11-to-22
31.31.31.31	11.11.11.11	Up	0	0	from-11-to-31

```
Total 4 displayed, Up 4, Down 0
```

PE12 also load balances the flows it receives to PE21 and PE22:

```
cse@PE12> clear mpls lsp statistics
```

```
cse@PE12> show mpls lsp statistics ingress
```

Ingress LSP: 4 sessions

To	From	State	Packets	Bytes	LSPname
11.11.11.11	12.12.12.12	Up	0	0	from-12-to-11
21.21.21.21	12.12.12.12	Up	5966	1527296	from-12-to-21
22.22.22.22	12.12.12.12	Up	5965	1527040	from-12-to-22
31.31.31.31	12.12.12.12	Up	0	0	from-12-to-31

Total 4 displayed, Up 4, Down 0

The traffic statistics on CE20 show that it receives the flows forwarded by PE21 and PE22 and delivers them to the destination host connected to port xe-0/0/20:

CE20		SSSeconds: 328		TTTime: 03:23:45	
Interface	Link	Input packets	(pps)	Output packets	(pps)
gr-0/0/0	Up	0	(0)	0	(0)
pfh-0/0/0	Up	0		0	
xe-0/0/0	Up	4216687111	(2000)	2692443334	(0)
xe-0/0/1	Up	3616302861	(2000)	4483879702	(0)
xe-0/0/2	Up	1531490139	(0)	1529917508	(0)
xe-0/0/20	Up	1411489270	(0)	1819919316	(3999)
xe-0/0/21	Up	1410356884	(0)	1409879658	(0)
xe-0/0/22	Up	1412630231	(0)	1412213416	(0)
xe-0/0/23	Up	1410405112	(0)	1661068651	(0)
xe-0/0/30	Up	196655	(0)	217656	(0)
ae0	Up	7832989972	(4000)	7176323036	(0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

The result is similar to the previous Layer 2 aliasing verification test as multiple levels of load balancing occur. Initially, CE10 load balances the traffic flows across the multi-homed uplinks to PE11 and PE12. The PEs then load balance traffic to the remote PEs in Data Center 2. What's different in this case is that the IRB interface on each ingress PE forwards the traffic using the route in its IPVPN-1 VRF. The route has two next hops, due to aliasing, and each ingress PE rewrites the Ethernet header and applies the EVPN and transport labels before forwarding.

As mentioned previously, additional levels of load balancing can take place. In this lab topology, for example, if there were multiple LSPs between PE11 and PE21 traffic flows would be further load balanced. The important point here is that bandwidth is efficiently utilized in all parts of the network for both Layer 2 and Layer 3 EVPN traffic flows.

## Inbound Routing from IP VPN Site

The integration of EVPN with IP VPN is also used to optimize the traffic paths of inbound traffic flows originating from sources outside the data centers destined to hosts or devices inside the data center. In this case the traffic reaches the data center via a remote IP VPN site. The source of the traffic could be at another intranet site of the customer or the Internet. These inbound traffic flows are always optimally routed to the data center PE closest to where the destination host resides, even if the destination host is a VM that has been migrated.

In the previous section we stepped through the actions an ingress PE takes when it learns of a host's MAC/IP address binding. Recall that, due to the placement of the EVPN VLAN's IRB interface in the IP VPN instance, the ingress PE installs a host route corresponding to the learned IP address in the IP VPN VRF with protocol type EVPN. It then transmits the host route to remote members of the IP VPN via a VPN route advertisement. An IP VPN PE router at a remote site receives the route advertisement and installs it in its local VRF with protocol type BGP. It is then able to route traffic directly to the data center PE closet to the data center host.

Routing gets tricky with workload migration because the VM's MAC/IP address and default gateway settings do not change. In the previous section we learned that Default Gateway Synchronization enables optimal routing of outbound traffic by the local data center PE after a VM has been migrated. Similarly, EVPN MAC mobility allows inbound traffic, from a remote host or device outside the data center to a migrated VM, to continue to flow along the most optimal data path.

For example, suppose a given data center host is a VM and is migrated to another data center. Once the migration event is complete the VM typically transmits a gratuitous ARP packet. A PE router at the destination data center receives this ARP and, due to ARP snooping, learns the MAC and IP address of the VM. This PE then updates its MAC-VRF and transmits a MAC/IP Advertisement route to all other EVPN PEs. It also updates the host route in its IP VPN VRF and transmits a VPN route to all other IP VPN PEs. The PE at the original data center site receives these route advertisements, updates its forwarding tables, and then withdraws its previously advertised MP-BGP routes corresponding to the host.

From the perspective of a remote non-data center IP VPN PE, it initially has a route to forward traffic destined to the VM to the PE in the original data center. After the VM migrates it receives a host route update from the PE in the destination data center. About a second later the route to the original data center is withdrawn. At the end of this



process the remote IP VPN PE has a host route for the VM pointing directly to the PE in the new data center. Thus, MAC Mobility is also applicable in this Layer 3 scenario as the data center PEs track the movement of the VM and inform the remote PE to forward inbound traffic to the VM using the most optimal path.

### Lab Example - Inbound and Outbound Routing with MAC Mobility

Initially, VM host 201.1.1.21 is actively running on Server 1 in Data Center 1. The MAC/IP binding of this host is discovered by PE12, which adds a host route in its local IP VPN VRF with protocol type EVPN. PE12 then transmits a VPN route advertisement to all remote PEs that are members of instance IPVPN-1:

```
cse@PE12> show route 201.1.1.21
```

```
IPVPN-1.inet.0: 23 destinations, 23 routes (23 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
201.1.1.21/32      *[EVPN/7] 00:16:40
                  > via irb.201
```

PE31 is located at a remote site and is a member of instance IPVPN-1. It receives the host route for 200.1.1.21 from PE12 via protocol BGP. Therefore, traffic from outside the data centers destined to 201.1.1.21 is forwarded directly to PE12 in Data Center 1:

```
cse@PE31> show route 201.1.1.21
```

```
IPVPN-1.inet.0: 15 destinations, 27 routes (15 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
201.1.1.21/32      *[BGP/170] 00:23:30, localpref 100, from 1.1.1.1
                  AS path: I, validation-state: unverified
                  > to 10.31.1.1 via xe-0/0/0.0, label-switched-path from-31-to-12
```

Looking at the route details on PE31 we see that it simply pushes the VPN label advertised by PE12 corresponding to its IPVPN VRF table, and then pushes a transport label to reach PE12:

```
cse@PE31> show route 201.1.1.21 detail
```

```
IPVPN-1.inet.0: 16 destinations, 28 routes (16 active, 0 holddown, 0 hidden)
```

```
201.1.1.21/32 (1 entry, 1 announced)
```

```
  *BGP      Preference: 170/-101
            Route Distinguisher: 12.12.12.12:121
            Next hop type: Indirect
            Address: 0x958948c
            Next-hop reference count: 17
            Source: 1.1.1.1
            Next hop type: Router, Next hop index: 590
            Next hop: 10.31.1.1 via xe-0/0/0.0, selected
```

```

Label-switched-path from-31-to-12
Label operation: Push 16, Push 300976(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 16: None; Label 300976: None;
Session Id: 0x1
Protocol next hop: 12.12.12.12

```

<snip>

Now we are ready to migrate the VM to Server 2 at Data Center 2 and verify that routing to and from a remote host or device continues to use the most optimal, direct path. First, let's start some traffic between the VM and a host at the remote site. From a CentOS Terminal window running on the VM, start a fast ping to Ixia 9/31's address 31.1.1.31. Note that the VM host is in VLAN 201 and is configured with a default gateway of 201.1.1.1, which matches the VLAN 201 IRB interface on the local Data Center 1 PEs, PE11 and PE12.

Monitoring the traffic on P1 shows that approximately 3000 pps of traffic is received on xe-0/1/0 from PE12 and forwarded to PE31 out interface xe-0/0/3. The ping response is received from PE31 and forwarded to PE12 out interface xe-0/1/0. Therefore, the initial outbound and inbound paths between Data Center 1 and the remote site are direct.

P1		Seconds: 1847		Time: 13:56:17	
Interface	Link	Input packets	(pps)	Output packets	(pps)
lc-0/0/0	Up	0		0	
pfh-0/0/0	Up	0		0	
xe-0/0/0	Up	1038550801	(6)	1046835151	(5)
xe-0/0/1	Down	0	(0)	0	(0)
xe-0/0/2	Down	0	(0)	0	(0)
xe-0/0/3	Up	1610660638	(3044)	1610672115	(3045)
lc-0/1/0	Up	0		0	
pfe-0/1/0	Up	0		0	
xe-0/1/0	Up	17862774	(3044)	9198189	(3044)
xe-0/1/1	Down	0	(0)	0	(0)
xe-0/1/2	Down	0	(0)	0	(0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

Next, let's move the VM host 201.1.1.21 to Server 2 in Data Center 2 using VMware vMotion. After a few seconds the host route learned by PE31 now points to PE22:

```
cse@PE31> show route 201.1.1.21
```

```

IPVPN-1.inet.0: 15 destinations, 27 routes (15 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

201.1.1.21/32      *[BGP/170] 00:01:11, localpref 100, from 1.1.1.1
                   AS path: I, validation-state: unverified
                   > to 10.31.1.1 via xe-0/0/0.0, label-switched-path from-31-to-22

```

Monitoring the traffic on P1 shows that the traffic is now received on xe-0/3/0 from PE22 and forwarded to PE31 out interface xe-0/0/3. The ping response is received from PE31 and forwarded to PE22 out interface xe-0/3/0. The new traffic pattern confirms that both the outbound and inbound paths remain direct.

P1		Seconds: 1986		Time: 13:58:36	
Interface	Link	Input packets	(pps)	Output packets	(pps)
lc-0/0/0	Up	0		0	
pfh-0/0/0	Up	0		0	
xe-0/0/0	Up	1038551918	(6)	1046836332	(6)
xe-0/0/1	Down	0	(0)	0	(0)
xe-0/0/2	Down	0	(0)	0	(0)
xe-0/0/3	Up	1611081112	(3106)	1611092586	(3106)
lc-0/1/0	Up	0		0	
pfe-0/1/0	Up	0		0	
xe-0/1/0	Up	18184596	(6)	9520063	(6)
xe-0/1/1	Down	0	(0)	0	(0)
xe-0/1/2	Down	0	(0)	0	(0)
xe-0/1/3	Down	0	(0)	0	(0)
lc-0/2/0	Up	0		0	
pfe-0/2/0	Up	0		0	
xe-0/2/0	Up	63876083	(6)	557310903	(6)
xe-0/2/1	Down	0	(0)	0	(0)
xe-0/2/2	Down	0	(0)	0	(0)
xe-0/2/3	Down	0	(0)	0	(0)
lc-0/3/0	Up	0		0	
pfe-0/3/0	Up	0		0	
xe-0/3/0	Up	501095727	(3108)	8188562	(3107)
xe-0/3/1	Down	0	(0)	0	(0)

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

For outbound traffic, PE22 is configured with a default gateway of 201.1.1.2 on VLAN 201. However, because of the EVPN Default Gateway Synchronization feature, PE22 routes the traffic on behalf of the 201.1.1.1 default gateway configured on PE11 and PE12 in Data Center 1:

```
cse@PE22> show bridge evpn peer-gateway-macs
```

```
Routing instance : EVPN-2
Bridging domain : V201, VLAN : 201
Installed GW MAC addresses:
00:00:c9:01:01:01
```

For inbound traffic, once the vMotion is complete the VM host transmits a gratuitous ARP, which is received, and snooped, by either PE21 or PE22 in Data Center 2. In this case, based on the observed forwarding path we know that PE22 receives the gratuitous ARP,

which triggers it to transmit both an EVPN MAC/IP advertisement route and an IP VPN host route update. PE31 receives the updated IP VPN host route and starts sending traffic destined to the VM to PE22. PE12, upon receiving the new MAC/IP route from PE22, updates its EVPN and IP VPN forwarding tables and withdraws the previously advertised EVPN and IPVPN routes corresponding to the VM host.

## MP-BGP EVPN Route Summary

As a reference, Table 3.1 below summarizes the EVPN specific MP-BGP route types and communities encountered during the verification process. For EVPN the MP-BGP NLRI packets use Address Family Identifier (AFI) of 25 (L2VPN) and a subsequent address family identifier (SAFI) of 70 (EVPN).

Table 3.1 Summary of EVPN MP-BGP Route Types

Route Type	Description	Extended Community
1 - Ethernet Auto-Discovery	The A-D per ESI route is used for fast convergence (MAC Mass Withdrawal) and Split Horizon filtering.  The A-D per EVI route is used for Aliasing.  These routes are only advertised when multi-homing is configured.	ESI Label Extended Community – advertised in the A-D per ESI route. Includes multi-homing mode (all-active or single-active) and Split Horizon label.
2 - MAC Advertisement	Advertises MAC and MAC/IP address reachability. Used for MAC learning/forwarding, MAC Mobility, Aliasing, Default Gateway Synchronization, and Asymmetric IRB Forwarding. Learned MAC/IP bindings generate EVPN host routes, which get added to IP VPN VRF for optimizing inbound routing to data center.	Default Gateway Extended Community – included when advertising MAC/IP of IRB.  MAC Mobility Extended Community – includes a sequence number that increments with each MAC move. Used by PE to correctly process MAC moves and to detect MAC flapping.
3 - Inclusive Multicast	Includes IM label, used when forwarding BUM traffic between PEs.	
4 - Ethernet Segment	For discovery of multi-homed neighbors and DF election. Only advertised when multi-homing is configured.	ES-Import Extended Community – value derived from ESI, used by receiving PE to filter incoming advertisements.

The next table summarizes the format of the MP-BGP EVPN routes received by a PE. These routes are all contained in the primary EVPN routing table and can be viewed on a PE using the `show route table bgp.evpn.0` command. Note that all routes start with the Route Type Identifier, values 1 through 4, and all have prefix length /304.

Table 3.2 Summary of EVPN MP-BGP Route Formats

Route Description	Route Example	Fields
1 - Ethernet Auto-Discovery per ESI	1:21.21.21.21:0::2222222222222222::FFFF:FF FF/304	<ul style="list-style-type: none"> <li>• Route Type “1”</li> <li>• RD unique to advertising PE</li> <li>• ESI</li> <li>• Ethernet Tag Id – reserved “0xFFFFFFFF”</li> </ul>
1 - Ethernet Auto-Discovery per EVI	1:21.21.21.21:1::2222222222222222::0/304	<ul style="list-style-type: none"> <li>• Route Type “1”</li> <li>• RD of advertising PE’s EVI</li> <li>• ESI</li> <li>• Ethernet Tag Id “0”</li> </ul>
2 - MAC Advertisement	2:21.21.21.21:1::100::00:00:09:c1:b0:d7/304	<ul style="list-style-type: none"> <li>• Route Type “2”</li> <li>• RD of advertising PE’s EVI</li> <li>• VLAN ID</li> <li>• MAC Address</li> </ul>
2 - MAC/IP Advertisement	2:21.21.21.21:1::100::00:00:09:c1:b0 :d7::100.1.1.29/304	<ul style="list-style-type: none"> <li>• Same as MAC Advertisement but includes host’s IP address</li> </ul>
3 - Inclusive Multicast	3:21.21.21.21:1::100::21.21.21.21/304	<ul style="list-style-type: none"> <li>• Route Type “3”</li> <li>• RD of advertising PE’s EVI</li> <li>• VLAN ID</li> <li>• Originator PE Loopback IP address</li> </ul>
4 - Ethernet Segment	4:12.12.12.12:0::1111111111111111:12.12.12 .12/304	<ul style="list-style-type: none"> <li>• Route Type “4”</li> <li>• RD unique to advertising PE</li> <li>• ESI</li> <li>• Originator PE Loopback IP address</li> </ul>



# Chapter 4

## High Availability Tests

In this chapter the resiliency of our now familiar, all-active, multi-homed EVPN network is tested. First, an access link between a data center PE and CE is failed and restored. Then we'll fail the PE node and power it back on. For each test case the recovery time of Layer 2 and Layer 3 traffic flows is measured using the IxNetwork application.

Specifically, the Ixia traffic generator is configured to transmit traffic flows between hosts at the data center and remote sites using the nine Ixia tester ports. A given flow from one port to another is transmitted at 1000 packets per second (pps) to make it easier to determine the recovery time of the flow when a change to the network topology occurs. For example, if 200 packets are lost for a particular flow during a failure event, then the flow's recovery time is 200 ms.

A summary of the test results is provided at the end of this chapter.

### Access Link

#### Link Failure

In this test case PE11's access interface xe-1/0/0 is physically disconnected and the impact to the traffic flows is measured and recorded.

#### Results

When the interface goes down the Status of the EVIs on PE11 also goes Down as seen in the CLI output for EVPN-1 below. This causes PE11 to withdraw all previously advertised EVPN and IP VPN routes.

The CLI output also shows that PE12 has become the DF as PE11's Ethernet Segment route for ESI 00:11:11:11:11:11:11:11:11 is withdrawn. PE11 also withdraws its Auto-Discovery per ESI and Auto-Discovery per EVI routes, which triggers the remote PEs in Data Center 2 to update the next hop for any MAC addresses destined to the ESI. PE11 then withdraws the individual MAC Advertisement routes:

```
cse@PE11> show evpn instance EVPN-1 extensive
Instance: EVPN-1
Route Distinguisher: 11.11.11.11:1
VLAN ID: 100
Per-instance MAC route label: 300944
MAC database status          Local Remote
Total MAC addresses:         0       3
Default gateway MAC addresses: 1       0
Number of local interfaces: 1 (0 up)
  Interface name  ESI                               Mode           Status
  ae0.100         00:11:11:11:11:11:11:11:11 all-active      Down
Number of IRB interfaces: 1 (1 up)
  Interface name  VLAN ID  Status  L3 context
  irb.100         100      Up      IPVPN-1
Number of bridge domains: 1
  VLAN ID  Intfs / up  Mode           MAC sync  IM route label
  100      1 0      Extended      Enabled
Number of neighbors: 3
12.12.12.12
  Received routes
  MAC address advertisement: 1
  MAC+IP address advertisement: 1
  Inclusive multicast: 1
  Ethernet auto-discovery: 2
21.21.21.21
  Received routes
  MAC address advertisement: 2
  MAC+IP address advertisement: 0
  Inclusive multicast: 1
  Ethernet auto-discovery: 2
22.22.22.22
  Received routes
  MAC address advertisement: 2
  MAC+IP address advertisement: 2
  Inclusive multicast: 1
  Ethernet auto-discovery: 2
Number of ethernet segments: 2
ESI: 00:11:11:11:11:11:11:11:11
Status: Resolved by NH 1048598
Local interface: ae0.100, Status: Down
Number of remote PEs connected: 1
  Remote PE      MAC label  Aliasing label  Mode
  12.12.12.12    300688     300688          all-active
Designated forwarder: 12.12.12.12
```



```

Advertised MAC label: 300976
Advertised aliasing label: 300976
Advertised split horizon label: 299984
ESI: 00:22:22:22:22:22:22:22:22:22
Status: Resolved by NH 1048609
Number of remote PEs connected: 2
  Remote PE      MAC label  Aliasing label  Mode
  21.21.21.21    300848        300848          all-active
  22.22.22.22    301040        301040          all-active

```

The IxNetwork statistics indicated the following results:

- All outbound traffic flows from Data Center 1 recovered the fastest. All of these flows are affected by the time it takes for CE10 to detect and switch over to using the link to PE12 exclusively. Layer 2 flows recovered within 109 ms as PE12 either forwards or floods traffic to Data Center 2. The IRB, or default gateway, interfaces on PE12 are configured the same as on PE11 which enabled the Layer 3 flows to recover within 116 ms.
- Inbound Layer 2 flows from Data Center 2 to Data Center 1 recovered within 345 ms. Prior to the failure it was observed that the PEs in Data Center 2 received MAC Advertisement routes for the Data Center 1 hosts from both Data Center 1 PEs. Therefore, when PE11 withdraws its previously advertised Auto-Discovery per ESI and MAC Advertisement routes, PE21 and PE22 update the next hop for destinations in Data Center 1 such that they point to PE12 only.
- Inbound Layer 3 flows originating from Data Center 2 and the Remote Site recovered within 1.17 seconds, and 2.19 seconds, respectively. Once the MAC-IP bindings contained in the EVPN MAC/IP and IP VPN advertisements are withdrawn by PE11, the host routes are no longer present in the IP VPN VRF. Traffic flows destined to hosts behind CE10 now traverse PE12, which ARPs for any unknown destinations. The ARP responses are snooped by PE12 and corresponding EVPN MAC/IP Advertisements and IP VPN host route updates are sent to all remote PEs. As the process of relearning a host route involves an ARP exchange and sending a route advertisement, traffic recovery for inbound Layer 3 traffic flows is comparatively slower than recovery for Layer 2 flows.

## Link Recovery

The flows are restarted on the Ixia and the link that was previously broken is restored. Impact to the traffic flows is then measured.

## Results

The PE11 xe-1/0/0 interface is configured with a hold-up timer of 180 seconds. This is to protect against packet loss upon node initialization (see the Node tests below) but is also invoked in this test case. During this time all traffic flows to/from Data Center 1 continue to flow through PE12.

Once the hold-up timer expires, LACP running between PE11 and CE10 ensures that both ends of the link are ready to transmit and receive traffic. This is important because small differences in the hold-up timers between PE11 and CE10 could cause packet loss. For example, CE10 may start sending traffic to PE11 before it is ready to receive it and vice versa.

Once the link comes up the EVIs on PE11 become active. PE11 re-advertises the ES, IM, and Auto-Discovery routes. This triggers a new DF election between PE11 and PE12. At the same time PE11 starts receiving traffic from its EVPN neighbors and CE10.

Results from the IxNetwork statistics showed the following:

- Routed traffic flows from Data Center 2 to Data Center 1 recovered within 144 ms as the next hops for destinations in Data Center 1 are updated in the IPVPN VRF on the PEs in Data Center 2.
- Routed traffic flows from Data Center 1 to Data Center 2 recovered in 1 ms.
- All other traffic flows are not impacted.

## Node

### Node Failure

The Ixia traffic flows are started and then PE11 is powered off to simulate a node failure. The impact to the traffic flows is then measured.

#### LAB NOTE

If you don't have physical access to the PE router, or you prefer not to power off your chassis, then the node failure can be simulated from the CLI. First, enter the shell as user root:

```
cse@PE11> start shell user root
Password:*****
root@PE11%
```

Then use the `ifconfig` command to bring each interface down. The best way to do this is to create a list of commands in a text editor, as shown below, and then paste them into the CLI session all at once. Make sure there is a carriage return after the last line:

```
ifconfig xe-1/0/0 down
ifconfig xe-1/2/0 down
ifconfig xe-1/1/0 down
ifconfig xe-2/0/0 down
ifconfig ae0 down
ifconfig ae1 down
```

When you are ready to bring the node back up repeat these commands, replacing the keyword *down* with *up*. This method can also be used for link failure and recovery testing.

## Results

There are a few mechanisms in the core layer of the topology that help minimize packet loss. First, when PE11 fails, any LSPs that terminated on PE11 are brought down and the respective head-end PEs are notified via RSVP Path Error messages. For both Layer 2 and Layer 3 traffic, the PEs in Data Center 2 remove the next hop LSP to PE11 and continue to forward to PE12 due to aliasing.

At the same time reachability, via OSPF, to PE11's loopback address is lost and the BFD timer for the MP-BGP session between PE11 and P1 expires after 600 ms. The P1 router subsequently withdraws all of the EVPN and IP VPN routes previously advertised by PE11. Similar to the link failure test case above, PE12 becomes the DF for its local ES.

The IxNetwork Statistics showed the following results:

- All affected outbound traffic flows from Data Center 1 are impacted by the time it takes for CE10 to detect and switch over to using the link to PE12 exclusively. Layer 2 flows recovered within 155 ms as PE12 either forwards or floods traffic to Data Center 2. For Layer 3 flows each of the multi-homed PE routers is configured with the same IP and MAC address such that flows are routed with minimum disruption, 155 ms in this case.
- Inbound Layer 2 flows from Data Center 2 to Data Center 1 recovered within 80 ms. Prior to the failure, the PEs in Data Center 2 received MAC Advertisements for all hosts in Data Center 1 from both PEs in Data Center 1. Thus, on PE21 and PE22 the next hop for destinations in Data Center 1 are updated to point to PE12 only.
- Inbound Layer 3 flows from Data Center 2 and the Remote Site recovered within 1.88 seconds and 876 ms respectively. These

flows took longer to recover than the inbound Layer 2 flows because the withdrawal of previously advertised EVPN MAC/IP and IP VPN host routes by P1 removes the routes from the IP VPN VRF. Traffic flows destined to hosts behind CE10 in Data Center 1 now traverse PE12, which ARPs for any unknown destinations. Upon snooping the ARP responses, PE12 sends EVPN MAC/IP Advertisements and IP VPN host route updates to all remote PEs.

## Node Recovery

The Ixia traffic flows are restarted and PE11 is powered on. The impact to the traffic flows is then measured.

### Results

The PE11 xe-1/0/0 and CE10 xe-0/0/0 interfaces are configured with a hold-up timer of 180 seconds, which is invoked once the link comes up. This setting is critically important in this scenario because it gives PE11 time to build its OSPF adjacencies, bring up the RSVP-TE LSPs, and establish the MP-BGP session to P1. During this time PE11 has awareness of its EVPN neighbors, although its neighbors are not aware of PE11 since there are no active ESIs, similar to the previous access link down test scenario. Without the hold-up timer CE10 would essentially forward traffic into a black hole for the amount of time it takes PE11 to complete initialization of all of its control protocols, approximately 2.5 minutes with this book's network configuration.

Once the hold-up timer expires, LACP running between PE11 and CE10 ensures that both ends of the link are ready to transmit and receive traffic. This is important because small differences in the hold-up timers between PE11 and CE10 could cause packet loss. For example, CE10 may start sending traffic to PE11 before it is ready to receive it and vice versa. Testing has shown that the use of LACP, even when there is a single link between the PE and CE, reduces packet loss significantly in this scenario.

Results from the IxNetwork statistics showed the following:

- Routed traffic flows from Data Center 1 to Data Center 2 recovered within 509 ms. Routed traffic flows from Data Center 1 to the Remote Site recovered within 18 ms. Before the access interface comes up the IRB interfaces on PE11 are down and the IP VPN VRF on PE11 only contains a route for the 31.1.1/24 network behind PE31 at the Remote Site. Once the access

interface is initialized the forwarding state in the VRF is populated and traffic is forwarded.

- Routed traffic flows from Data Center 2 to Data Center 1 recovered in 354 ms. Once the access interface on PE11 comes up the PEs in Data Center 2 receive EVPN updates from PE11 and update the entries in their IP VPN VRFs to utilize the new next hop.
- Layer 2 traffic flows are minimally impacted, 2 ms outbound to Data Center 2 and 55 ms inbound from Data Center 2.
- All other traffic flows are not affected.

## High Availability Test Summary

The following tables summarize the worst-case packet loss for each high availability test. The results are categorized by traffic type, Layer 2 versus Layer 3, by traffic direction, inbound versus outbound, and by site, data centers and the remote site.

Table 4.1 Summary of High Availability Test Results

Test Case	DC1 Outbound L2 Flows to DC2	DC1 Inbound L2 Flows from DC2	DC1 Outbound L3 Flows to DC2	DC1 Inbound L3 Flows from DC2	DC1 Outbound L3 Flows to Remote Site	DC1 Inbound L3 Flows from Remote Site
Access Link Failure	109 ms	345 ms	116 ms	1.17 sec	109 ms	2.19 sec
Access Link Recovery	0	0	1 ms	144 ms	0	0
Node Failure	155 ms	80 ms	155 ms	1.88 sec	155 ms	876 ms
Node Recovery	2 ms	55 ms	509 ms	354 ms	18 ms	0

## Conclusion

The Proof of Concept testing of EVPN in Juniper's POC Labs demonstrates its applicability for use as a DCI technology. The control plane-based learning of MAC addresses enables many significant features such as all-active multi-homing for increased resilience and traffic load balancing, as well as MAC mobility. The seamless integration of routing capabilities provides efficient forwarding of inbound and outbound traffic flows on the most optimal path, even when a host is migrated from one data center to another. Finally, the high availability testing shows that the solution is resilient and recovers quickly upon a link and node failure and restoration events.

**REMEMBER** The configuration files for all devices used in this POC Lab *Day One* book can be found on this book's landing page at <http://www.juniper.net/dayone>. The author has also set up a Dropbox download for those readers not logging onto the *Day One* website, at: <https://dl.dropboxusercontent.com/u/18071548/evpn-configs.zip>. Note that this URL is not under control of the author and may change over the print life of this book.